



Titre: Commande par microprocesseurs de redresseurs polyphasés
Title:

Auteur: Adrian-Valentin Fadei
Author:

Date: 1996

Type: Mémoire ou thèse / Dissertation or Thesis

Référence: Fadei, A.-V. (1996). Commande par microprocesseurs de redresseurs polyphasés
Citation: [Master's thesis, École Polytechnique de Montréal]. PolyPublie.
<https://publications.polymtl.ca/6721/>

 **Document en libre accès dans PolyPublie**
Open Access document in PolyPublie

URL de PolyPublie: <https://publications.polymtl.ca/6721/>
PolyPublie URL:

**Directeurs de
recherche:**
Advisors:

Programme: Unspecified
Program:

UNIVERSITÉ DE MONTRÉAL

**COMMANDE PAR MICROPROCESSEURS
DE REDRESSEURS POLYPHASÉS**

ADRIAN-VALENTIN FADEI

DÉPARTEMENT DE GÉNIE ÉLECTRIQUE ET GÉNIE INFORMATIQUE
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(GÉNIE ÉLECTRIQUE ET INFORMATIQUE)

AOÛT 1996

@ Adrian-Valentin Fadei, 1996



**National Library
of Canada**

**Acquisitions and
Bibliographic Services**

**395 Wellington Street
Ottawa ON K1A 0N4
Canada**

**Bibliothèque nationale
du Canada**

**Acquisitions et
services bibliographiques**

**395, rue Wellington
Ottawa ON K1A 0N4
Canada**

Your file Votre référence

Our file Notre référence

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-33130-X

UNIVERSITÉ DE MONTRÉAL
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé:

COMMANDE PAR MICROPROCESSEURS
DE REDRESSEURS POLYPHASÉS

présenté par: FADEI Adrian-Valentin

en vue de l'obtention du diplôme de: Maîtrise ès sciences appliquées

a été dûment accepté par le jury d'examen constitué de:

M. Deschênes J.-G., M.Sc.A. , président

M. Olivier G., Ph.D., membre et directeur de recherche

M. April G.-E., M.Sc., membre et codirecteur de recherche

M. Roy G., M.Sc.A., membre

À mes parents Lidia et Nichita
et ma fiancée Aïda.

REMERCIEMENTS

L'auteur de ce mémoire tient à remercier toutes les personnes qui ont contribué de près ou de loin à la réalisation de ce projet. Plus particulièrement, ces remerciements s'adressent à MM. G. Olivier et G.-E. April pour avoir proposé et dirigé ce projet. L'auteur tient aussi à remercier M. Jacques Girardin pour la préparation du matériel.

SOMMAIRE

De nos jours, dans les liens HTCC on utilise des redresseurs à haute puissance. Les grandes quantités d'énergie transformées justifient les recherches théoriques menées dans le domaine des nouveaux convertisseurs de courant à la section électrotechnique de l'École Polytechnique de Montréal. Ces recherches visent à améliorer le rendement et la qualité de la conversion du courant alternatif en courant continu.

Une des solutions envisagées consiste à utiliser des redresseurs à grand nombre de phases. Cette approche se caractérise par un ordre plus élevé des harmoniques de la tension de sortie, ce qui implique l'utilisation de filtres moins volumineux. En plus, l'utilisation des redresseurs à grand nombre de phases réduit l'amplitude des harmoniques de courant injectés dans le réseau.

En conséquence, on a réalisé dans le cadre de cette maîtrise un système de commande par micro-ordinateur utilisable pour différentes configurations de redresseurs. Le système réalisé permettra aux recherches théoriques et aux simulations par ordinateur de franchir une nouvelle étape, celle des évaluations expérimentales. La commande des redresseurs, basée sur un micro-ordinateur,

permet de commander plusieurs configurations de redresseurs en partant de quatre ponts Graetz indépendants, jusqu'à des configurations série et parallèle de redresseurs à vingt-quatre phases. Le micro-ordinateur est aussi capable de mesurer cinq tensions et cinq courants tout en assurant les protections du système; son logiciel de commande, écrit en Forth et en Assembleur, réalise un réglage de la tension de sortie et la génération des commandes des thyristors par verrouillage de phase des angles d'allumage.

ABSTRACT

Industry in general and HVDC lines use high power rectifiers. Intense research are done on new rectifiers topologies at École Polytechnique de Montréal. The goal of this research is to improve electrical parameters associated with the AC-DC conversion in high power rectifiers.

One of the directions investigated in these researches are high pulse number converters. This solution let reduce harmonics contents in the output voltage as well as in the AC line currents.

In order to improve our investigations and to make better possible experimental evaluations, a multiprocessor control system for thyristor converter was developed. This system can control configurations from six up to twenty-four thyristors. The interface board also allows measuring of five voltages and five currents, and provides protection features for the microprocessor system. The microprocessor software, written in Forth and Assembler, control the thyristor phase angle and regulates the DC output voltage.

TABLE DES MATIÈRES

Dédicace	IV
Remerciements	V
Sommaire	VI
Abstract	VIII
Table des matières	IX
Liste des tableaux	XII
Liste des figures	XIII
Liste des notations et des symboles	XVI
Liste des annexes	XVII
1. Introduction	1
2. Configurations de l'électronique de puissance	4
2.1. Redresseurs utilisant un seul micro-contrôleur esclave	7
2.1.1. Redresseur double étoile triphasée avec transformateur d'interphase.....	8
2.1.2. Redresseur hexaphasé à neutre isolé	10
2.1.3. Pont de Graetz	11
2.1.4. Pont triphasé hybride	14
2.2. Redresseurs utilisant plusieurs micro-contrôleurs esclaves	14
2.2.1. Redresseur dodécaphasé à transformateur d'interphase	14

2.2.2. Connexion en parallèle des ponts	16
2.2.3. Connexion en série des ponts	19
3. Circuit d'interface entre le micro-ordinateur et l'étage de puissance	23
3.1. Architecture du micro-ordinateur Conforth.....	23
3.2. Circuit d'interface du micro-contrôleur maître	26
3.2.1. Mesure de la tension du réseau	27
3.2.2. Mesure du courant	29
3.2.3. Détection de la variation des conditions extérieures	35
3.2.4. L'affichage lumineux de l'état du système	37
3.3. Circuit d'interface du micro-contrôleur esclave	39
3.3.1. Mesure de la tension de sortie	40
3.3.2. Mesure du courant de sortie	43
3.3.3. Commande des thyristors	43
4. Logiciel de commande	52
4.1. Logiciel du micro-contrôleur maître	53
4.1.1. Sous-programmes de communication	55
4.1.2. Sous-programmes de mesure et de protection	61
4.1.3. Sous-programmes de signalisation	68
4.1.4. Sous-programmes de commande	68
4.2. Sous-programmes des micro-contrôleurs esclaves.....	73
4.2.1. Sous-programmes de communication	76
4.2.2. Sous-programmes de mesure	79

4.2.2. Sous-programme de commande	82
4.3. Réponse du système de réglage à une entrée de type échelon.....	83
5. Conclusion	91
Bibliographie	93
Annexes	94

LISTE DES FIGURES

Figure 2.1	Connexion des circuits de détection des passages par zéro	5
Figure 2.2	La forme d'onde généralisée pour la tension de sortie	7
Figure 2.3	Double étoile triphasée avec transformateur d'interphase....	9
Figure 2.4	Redresseur hexaphasé à neutre isolé	10
Figure 2.5	Pont de Graetz	11
Figure 2.6	Commande des thyristors d'un pont de Graetz	12
Figure 2.7	Quatre ponts de Graetz indépendants	13
Figure 2.8	Redresseur dodécaphasé à transformateur d'interphase	15
Figure 2.9	Redresseur à vingt-quatre phases - connexion parallèle	17
Figure 2.10	Redresseur à vingt-quatre phases - connexion série	20
Figure 3.1	Architecture du système et flux de données du système.....	25
Figure 3.2	Interface du micro-contrôleur maître: circuit de mesure de la tension	28
Figure 3.3	Circuit de mesure du courant: formes d'ondes	30
Figure 3.4	Interface du micro-contrôleur maître: circuit de mesure du courant	31
Figure 3.5	Détection de la variation du courant de sortie: formes d'ondes	36
Figure 3.6	Interface du micro-contrôleur maître: circuits divers	38

Figure 3.7	Interface du micro-contrôleur esclave:	
	circuit de mesure de la tension de sortie	41
Figure 3.8	Mesure de la tension de sortie: formes d'ondes	42
Figure 3.9	Interface du micro-contrôleur esclave:	
	circuit de mesure du courant de sortie	44
Figure 3.10	Mesure de l'angle de commande: formes d'ondes	46
Figure 3.11	Commande des thyristors: formes d'ondes	49
Figure 3.12	Interface du micro-contrôleur esclave:	
	circuit de commande des thyristors	50
Figure 4.1	Architecture du programme du micro-contrôleur maître	54
Figure 4.2	Structure de la matrice « état »	56
Figure 4.3	Identification des données à transmettre	57
Figure 4.4	Configuration d'une trame de données	57
Figure 4.5	Registre d'état et de contrôle	59
Figure 4.6	Registre secondaire du bus maître - esclave	60
Figure 4.7	Protection de courant du micro-contrôleur maître.....	62
Figure 4.8	Disparition d'une phase d'alimentation:	
	protection réalisée par le programme principal.....	64
Figure 4.9	Disparition d'une phase d'alimentation:	
	protection basée sur la détection par le matériel de la	
	variation de la tension du réseau.....	65

Figure 4.10	Calcul de la consigne des micro-contrôleurs esclaves	72
Figure 4.11	Architecture du logiciel du micro-contrôleur esclave	74
Figure 4.12	Commande équidistante des thyristors	75
Figure 4.13	Réponse de la boucle d'angle à une consigne de type échelon	84
Figure 4.14	Réponse des boucles de tension à une consigne de type échelon pour la commande de quatre ponts indépendants.	85
Figure 4.15	Réponse à une consigne de type échelon des boucles d'angles d'un redresseur dodécaphasé.....	87
Figure 4.16	Réponse à une consigne de type échelon de la boucle de tension d'un redresseur dodécaphasé.....	88
Figure 4.17	Réponse à une consigne de type échelon des boucles d'angle d'un redresseur à vingt-quatre phases	89
Figure 4.18	Réponse à une consigne de type échelon de la boucle de tension d'un redresseur à vingt-quatre phases	90

LISTE DE TABLEAUX

Tableau 3.1: Signaux du connecteur d'entrées/sorties du μ P M 37700 ...	34
Tableau 3.2: Signaux du connecteur d'entrées/sorties du μ P M 37450 ...	47
Tableau A.1: Contenu de la matrice de variables réelles « état »	94
Tableau B.1: Contenu de la matrice d'entiers « b_état »	97

LISTE DES SYMBOLES ET INDICES

α	angle de commande des thyristors (degrés électriques).
ω	pulsation de la tension du réseau (rad s^{-1}).
a_{initial}	angle initial de la commande des thyristors (degrés électriques).
a_{final}	angle final de la commande des thyristors (degrés électriques).
E	tension efficace de phase pour une connexion étoile ou la tension efficace de ligne pour une connexion en pont (V).
E_o	tension continue de sortie du redresseur (V).
$E_{o,i}$	tension de sortie du pont i (V).
E_{ll}	tension efficace de ligne (V).
E_{ph}	tension efficace de phase (V).
I_o	courant continu de sortie du redresseur (A).
$I_{o\text{ eff}}$	courant efficace de sortie d'un redresseur (A).
q	indice de phases.
p	indice de pulsations à la sortie d'un convertisseur.
U	tension (V).
μP	abréviation pour microprocesseur.
μC	abréviation pour micro-contrôleur.

LISTE DES ANNEXES

Annexe A	94
Annexe B	97
Annexe C	99
Annexe D	101
Annexe E	115
Annexe F	133

1. INTRODUCTION

Les grandes quantités d'énergie électrique transportées sur de longues distances et la croissance du nombre des consommateurs de grande taille nécessitent l'optimisation de la conversion de l'énergie électrique. L'utilisation des nouvelles configurations de redresseurs commandés apporte une contribution intéressante à la réduction des coûts de conversion de l'énergie électrique en améliorant la qualité de l'énergie fournie aux consommateurs.

Le professeur Guy Olivier de la section électrotechnique de l'École Polytechnique de Montréal en collaboration avec le professeur G.-E. April de la section électronique dirige des recherches très actives dans le domaine des nouvelles topologies de convertisseurs de courant. Les études théoriques et les simulations par ordinateur des nouvelles configurations de convertisseurs doivent être complétées par des évaluations expérimentales.

Par conséquent, l'objectif de cette maîtrise a été de réaliser un système de commande par micro-ordinateur qui s'adapte facilement aux différentes configurations de redresseurs. Le système réalisé permet la commande de plusieurs configurations de redresseurs en partant de quatre ponts de Graetz indépendants, jusqu'à des configurations de redresseurs à vingt-quatre phases.

En plus, le micro-ordinateur mesure cinq tensions et cinq courants et permet d'assurer les fonctions de régulation et de protection du système de commande.

Le système de commande est constitué des cartes Conforth et Redforth. La carte Conforth a été développée sous la direction du professeur G.-E. April à l'École Polytechnique de Montréal [1] et elle contient cinq microprocesseurs qui forment une architecture maître - quatre esclaves. Le microprocesseur maître calcule la consigne d'angle et la transmet aux micro-contrôleurs esclaves qui commandent les thyristors et mesurent les tensions et les courants de sortie. La carte Redforth, qui a été développée dans le cadre de cette maîtrise, sert d'interface entre le système de commande et l'électronique de puissance en permettant la commande des thyristors, l'acquisition des tensions et des courants. Elle assure aussi le matériel pour les protections du système.

Dans le deuxième chapitre nous allons exposer les principales configurations de l'électronique de puissance commandables par le système développé. Nous montrerons les schémas du circuit électronique de puissance, les connexions à la carte d'interface. Pour chaque configuration nous donnerons le courant maximal de charge admissible et la tension de sortie continue, fournie par le redresseur pour un angle de commande nul.

Le troisième chapitre présentera la carte Redforth qui constitue l'interface entre le micro-contrôleur et l'électronique de puissance. La carte a été conçue à

l'aide d'un logiciel de conception de circuit imprimé (ORCAD). Nous allons montrer les schémas électriques, les principaux points d'ajustement des circuits et nous allons expliquer le fonctionnement des circuits électroniques.

Le chapitre quatre a pour objet le logiciel de commande des microprocesseurs maître et esclaves. Le logiciel est écrit en Forth et Assembleur et utilise les compilateurs présents dans les mémoires programmables des micro-contrôleurs maître et esclaves. Le programme contient des tâches en temps réel qui s'exécutent simultanément sur les micro-contrôleurs maître et esclaves. L'échange des données entre les micro-contrôleurs permet de fermer les boucles de réglage d'angle et de tension. En plus, la carte communique avec un ordinateur personnel qui réalise l'affichage de l'état du système. Finalement, le système de commande est testé pour plusieurs configurations utilisant deux et quatre micro-contrôleurs esclaves.

Le dernier chapitre constitue la conclusion qui découle de la réalisation et des tests sur le système de commande de convertisseurs.

2. CONFIGURATIONS DU CIRCUIT ÉLECTRONIQUE DE PUISSANCE

Le deuxième chapitre présente les configurations de redresseurs pour lesquels on a développé une partie du matériel et le logiciel de commande. Les configurations présentées dans ce chapitre sont commandées par un, deux ou quatre micro-contrôleurs esclaves, en fonction de la complexité du montage de l'électronique de puissance. Ainsi, chaque micro-contrôleur esclave peut commander six thyristors, ce qui représente l'équivalent d'un pont de Graetz ou de deux étoiles triphasées. En plus, chaque micro-contrôleur mesure la tension et le courant de sortie qui sont fournis par les diviseurs de tension « div. i » où $i = 0...3$ et les shunts de courant « sh. i » où $i = 0... 3$ qui ne sont pas inclus dans la carte d'interface. La mesure du courant de sortie par les micro-contrôleurs esclaves assure la protection lente des thyristors. La protection rapide des thyristors est réalisée par le micro-contrôleur maître, qui mesure le courant du shunt « sh.M » connecté en série avec la charge du redresseur. La mesure de la tension de sortie par les micro-contrôleurs esclaves contribue au réglage de la tension.

Les microprocesseurs esclaves synchronisent les commandes des thyristors à la tension secondaire des transformateurs.

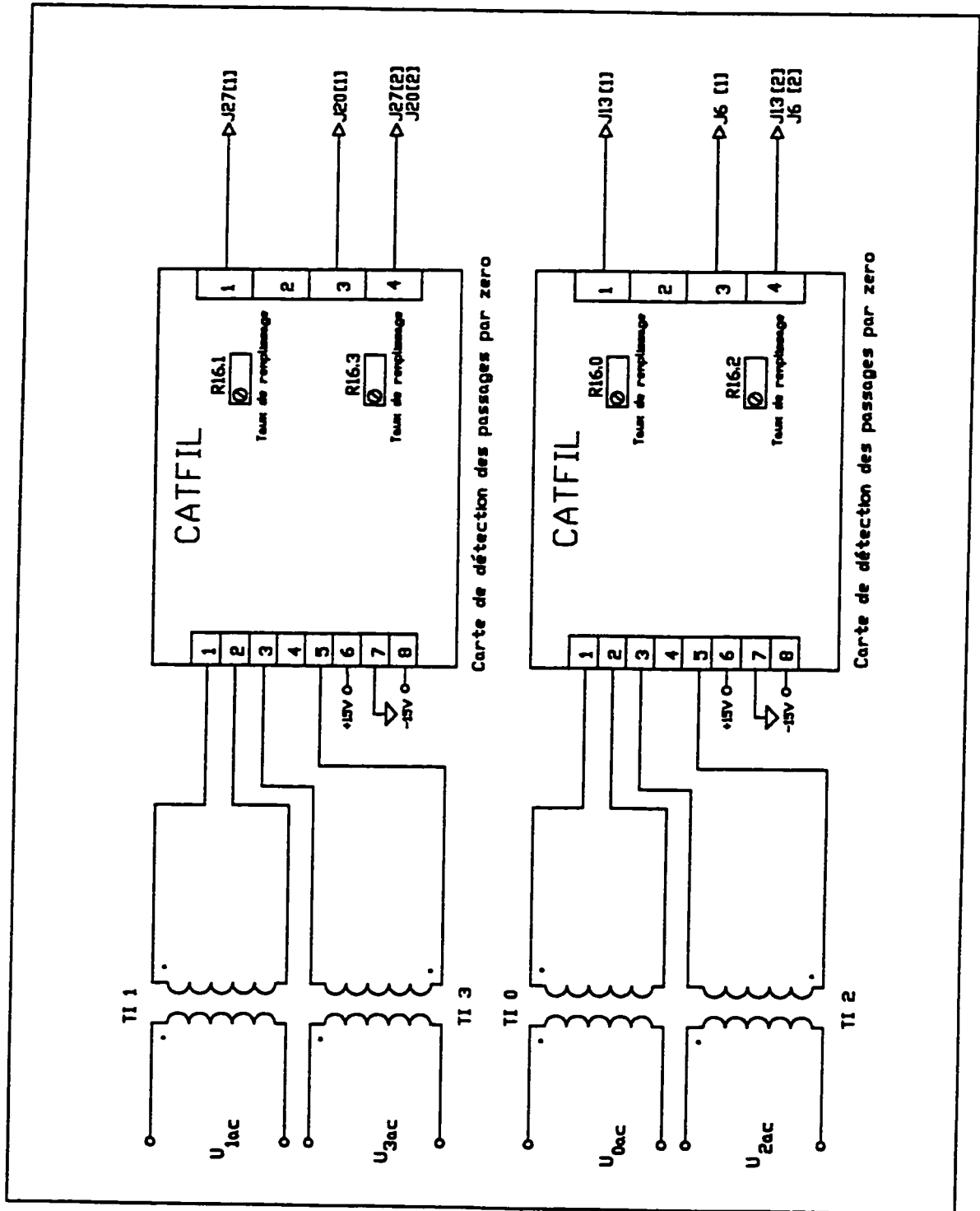


Figure 2.1 Connexion des circuits de détection des passages par zéro.

Le signal de synchronisation est fourni à chaque esclave par les cartes CATFIL [2], réalisées au Département de Génie Électrique de l'École Polytechnique de Montréal, qui détectent le passage par zéro de la tension de ligne U_{ca} . La figure 2.1 indique la connexion de ces cartes.

Chaque redresseur se caractérise par la tension continue de sortie et par le courant maximal de charge. Les formules présentées dans ce chapitre se basent sur les hypothèses suivantes:

- On néglige les chutes de tension sur les thyristors, l'empiétement pendant la commutation ainsi que l'impédance de sortie du transformateur.
- Une charge fortement inductive lisse le courant de charge. Le courant continu de sortie d'un pont est calculé en utilisant le courant moyen en conduction directe des thyristors ($I_{F(AV)}$), selon la configuration choisie. Les thyristors IRKT 56 utilisés dans l'étage de puissance ont un courant moyen maximal $I_{F(AV)} = 55$ A.

La figure 2.2 présente la tension de sortie généralisée d'un redresseur. La tension continue de sortie (2.1) s'obtient par intégration de l'aire sous la courbe de la tension de sortie.

$$E_o = \frac{1}{2\pi} \int_{\frac{-\pi}{k} + \alpha}^{\frac{+\pi}{k} + \alpha} E\sqrt{2} * \cos(\omega * t) = E\sqrt{2} * \frac{\sin\left(\frac{\pi}{k}\right)}{\frac{\pi}{k}} * \cos \alpha ; \quad (2.1)$$

où E représente la tension efficace de phase pour une connexion étoile ou la tension efficace de ligne pour une connexion pont. Selon la configuration de l'étage de puissance, k est égal à l'indice de pulsation de la tension de sortie (p) pour un redresseur connecté en pont ou à l'indice de phase (q) pour un redresseur utilisant une connexion étoile. ω représente la pulsation de la tension du réseau alors que α indique l'angle de commande des thyristors.

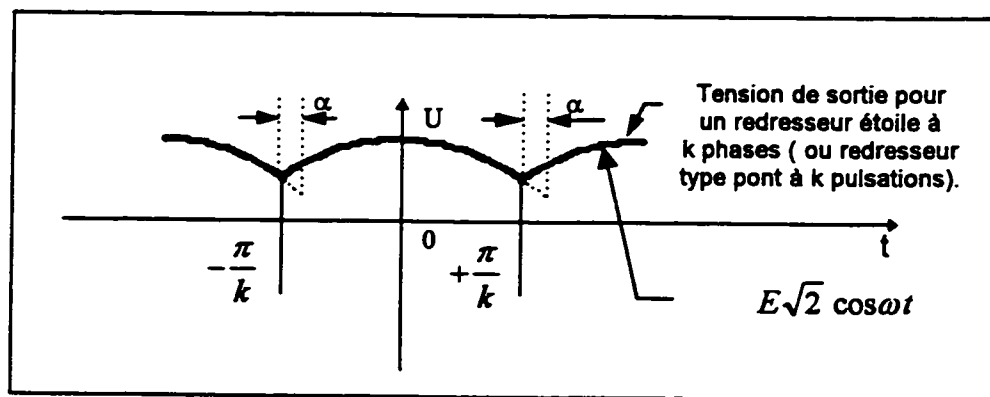


Figure 2.2 La forme d'onde généralisée pour la tension de sortie

Les thyristors utilisées dans l'étage de puissance déterminent le courant de sortie maximal. Nous allons indiquer les valeurs maximales de la tension et du courant de sortie pour chaque configuration présentée.

2.1 Redresseurs utilisant un seul micro-contrôleur esclave

Le micro-ordinateur permet la commande simultanée ou indépendante de quatre redresseurs triphasés que nous allons présenter dans cette section.

2.1.1 Redresseur double étoile triphasé avec transformateur d'interphase

La figure 2.3 présente le schéma d'un redresseur en double étoile triphasée avec transformateur d'interphase. Les six enroulements secondaires du transformateur d'alimentation sont groupés en deux étoiles distinctes. Le transformateur d'interphase raccorde les points neutres N1 et N'1 des secondaires. Le fonctionnement normal en charge est atteint lorsque le courant de charge dépasse habituellement $I_o \cong \frac{0.005E_s}{L_b\omega}$; [3], ce qui permet au transformateur d'interphase d'équilibrer les courants de charge fournis par les deux étoiles. La tension continue de sortie est définie en fonction de la tension de phase E_{ph} par :

$$E_o = \frac{3\sqrt{3}}{\pi\sqrt{2}} E_{ph} \cos \alpha = 1,17 E_{ph} \cos \alpha; \quad (2.2)$$

Le courant de sortie du redresseur dépend du courant moyen des thyristors:

$$I_o \propto 6 I_{F(AV)}; \quad (2.3)$$

Lors de la connexion des secondaires, une attention particulière doit être portée au sens des enroulements. Une connexion incorrecte entraîne la saturation du transformateur d'interphase.

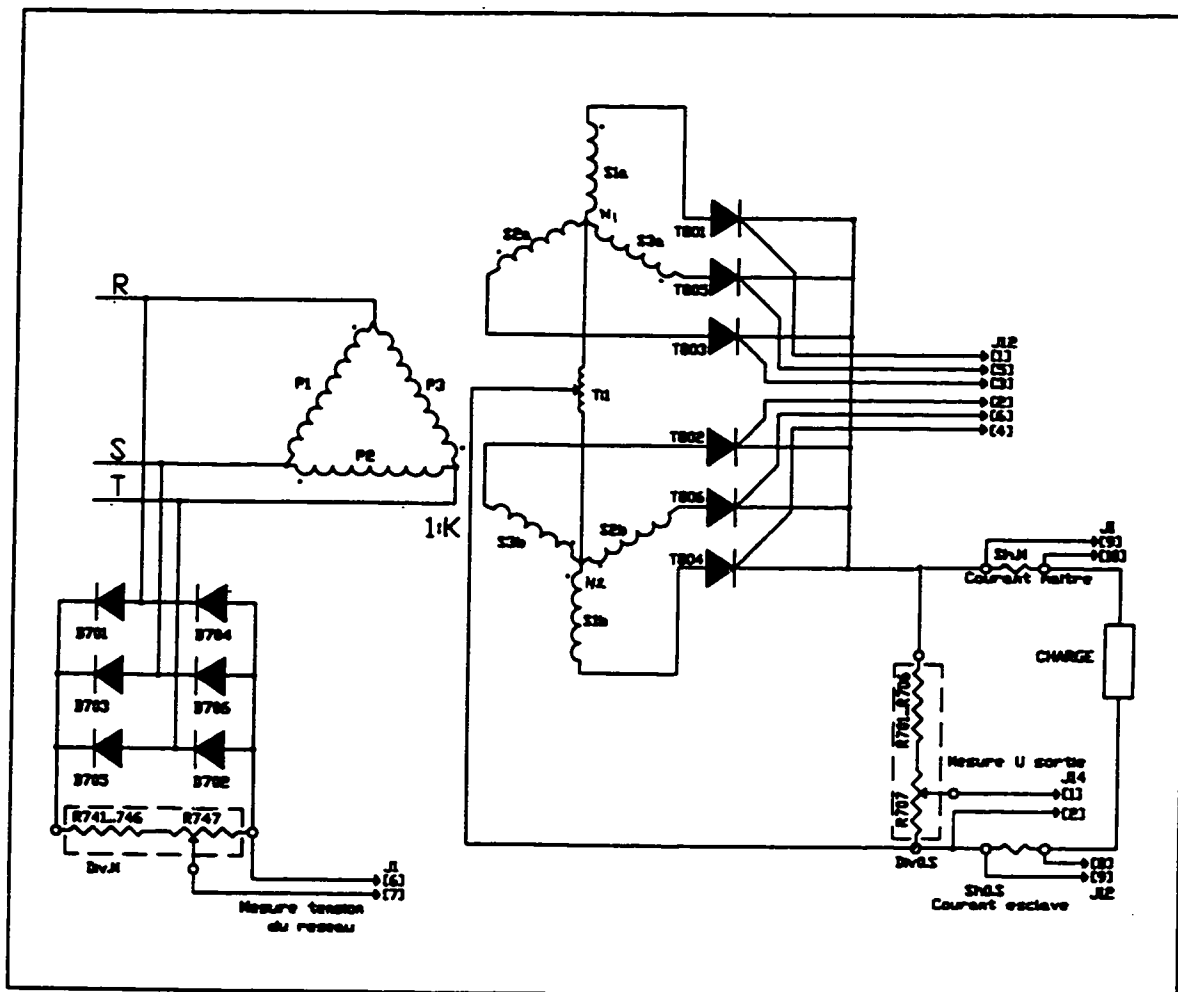


Figure 2.3 Double étoile triphasée avec transformateur d'interphase.

La commande des thyristors dans cette configuration est similaire à celle d'un pont de Graetz.

2.1.2 Redresseur hexaphasé à neutre isolé

Dans un redresseur hexaphasé, les six enroulements du secondaire du transformateur sont connectés à un point commun N (fig. 2.4).

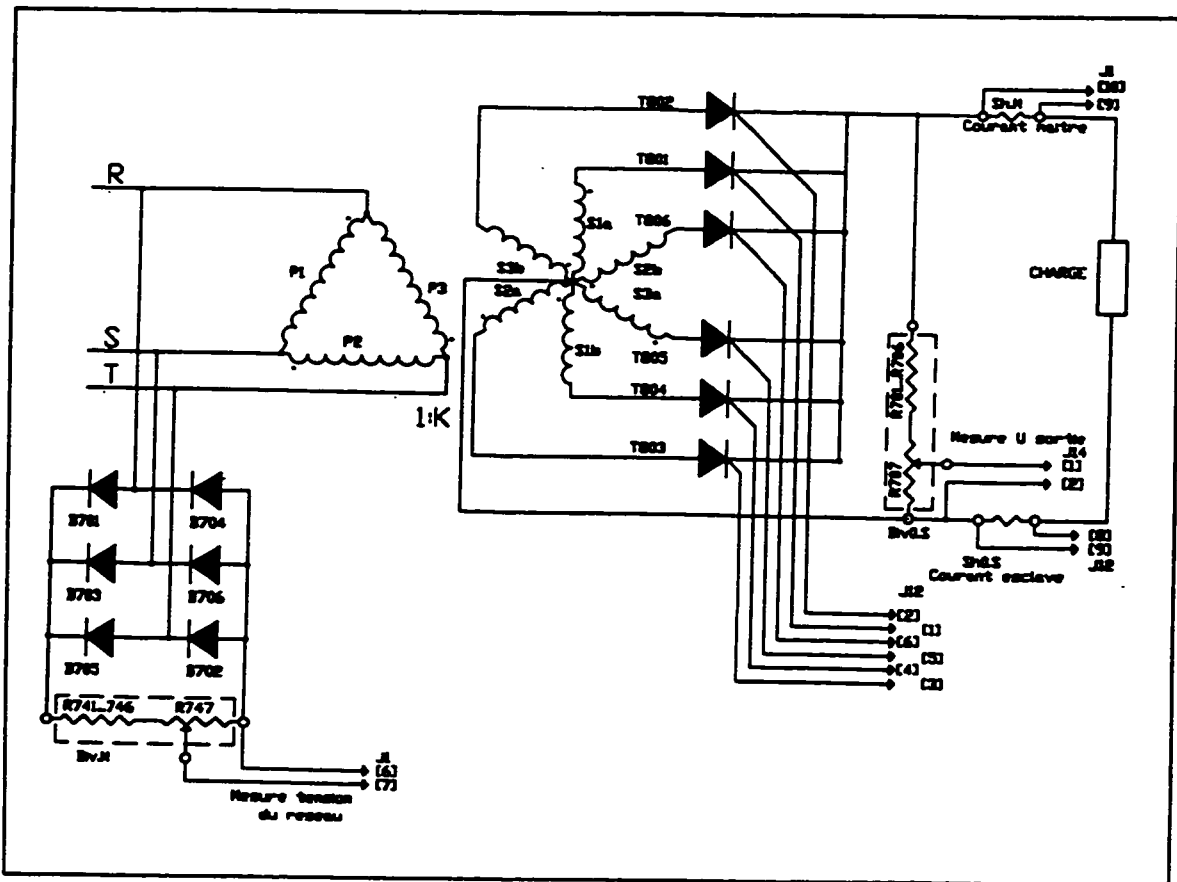


Figure 2.4 Redresseur hexaphasé à neutre isolé

La tension de sortie du redresseur E_{∞} se calcule de la même façon que celle de la section 2.1.1.

Le courant efficace d'un thyristor détermine le courant de charge maximal du redresseur:

$$I_{o\infty} = 3 I_{P(AV)} ; \quad (2.4)$$

La commande des thyristors est similaire à celle d'un pont de Graetz que nous allons présenter dans la section suivante.

2.1.3 Pont de Graetz

La figure 2.5 présente le redresseur en pont triphasé . La tension continue de sortie de cette configuration est donnée par :

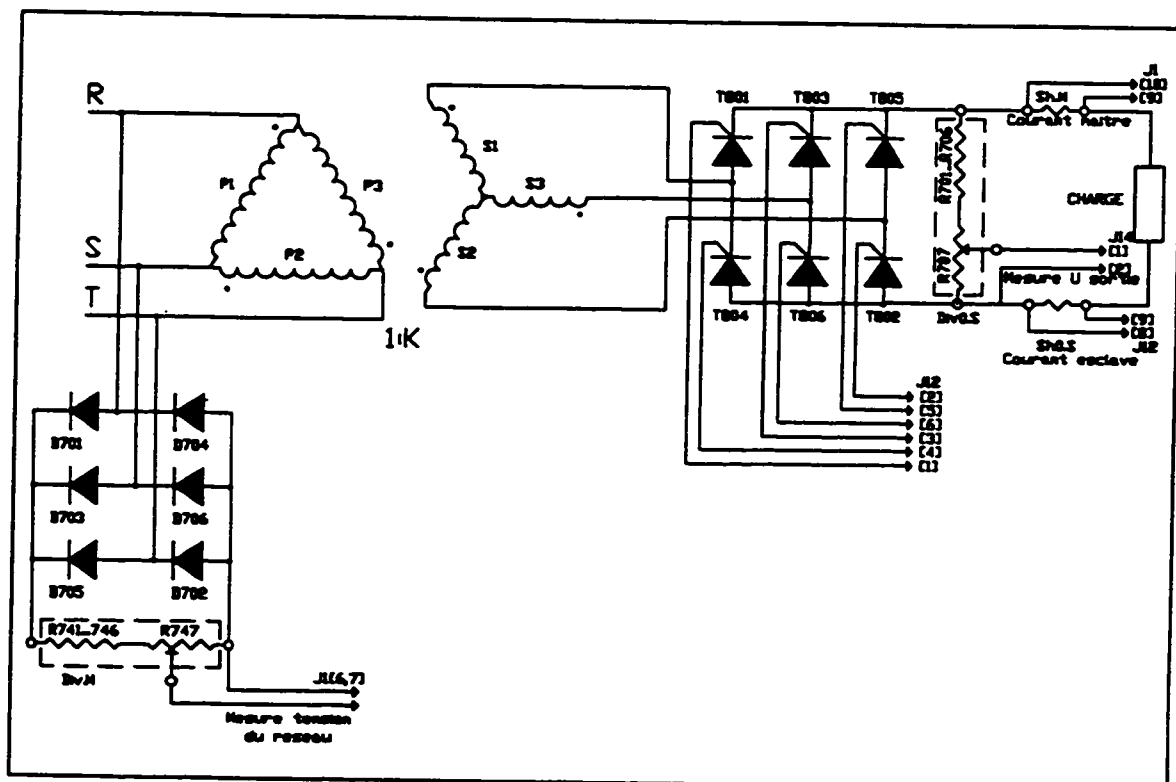


Figure 2.5 Pont de Graetz

$$E_o = \frac{3\sqrt{2}}{\pi} E_u \cos \alpha = 1,35 E_u \cos \alpha ; \quad (2.5)$$

où E_u correspond à la tension de ligne et α est l'angle de commande.

Le courant maximal de charge déterminé par le courant efficace des thyristors est:

$$I_{o\alpha} = 3 I_{F(AV)}; \quad (2.6)$$

La figure 2.6 présente la séquence de commande des thyristors.

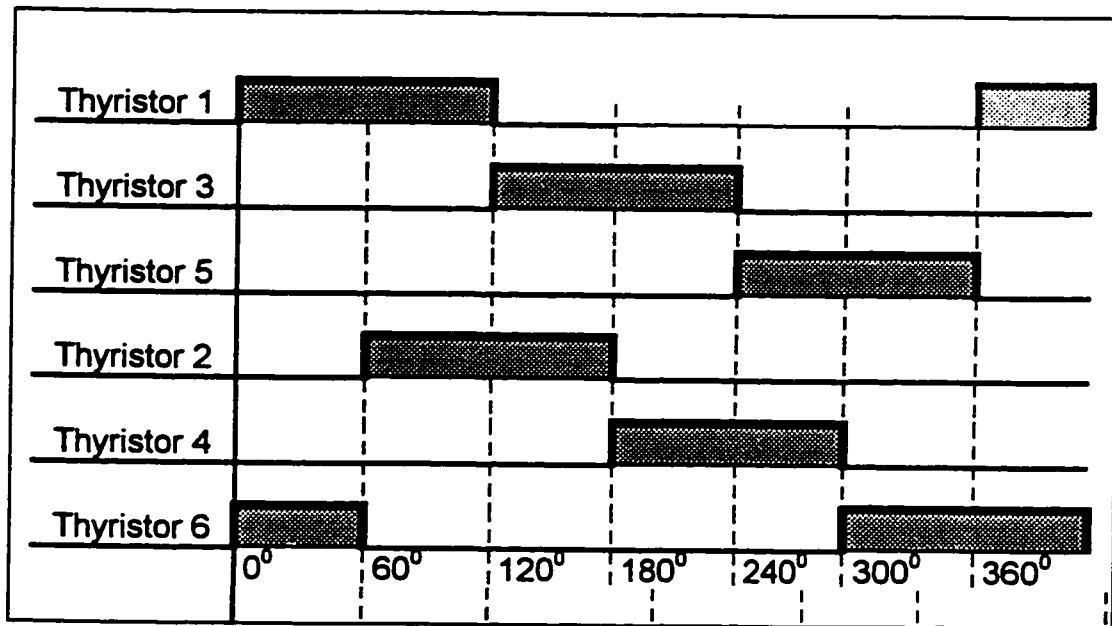


Figure 2.6 Commande des thyristors d'un pont de Graetz

La figure 2.7 montre la connexion de quatre ponts de Graetz indépendants.

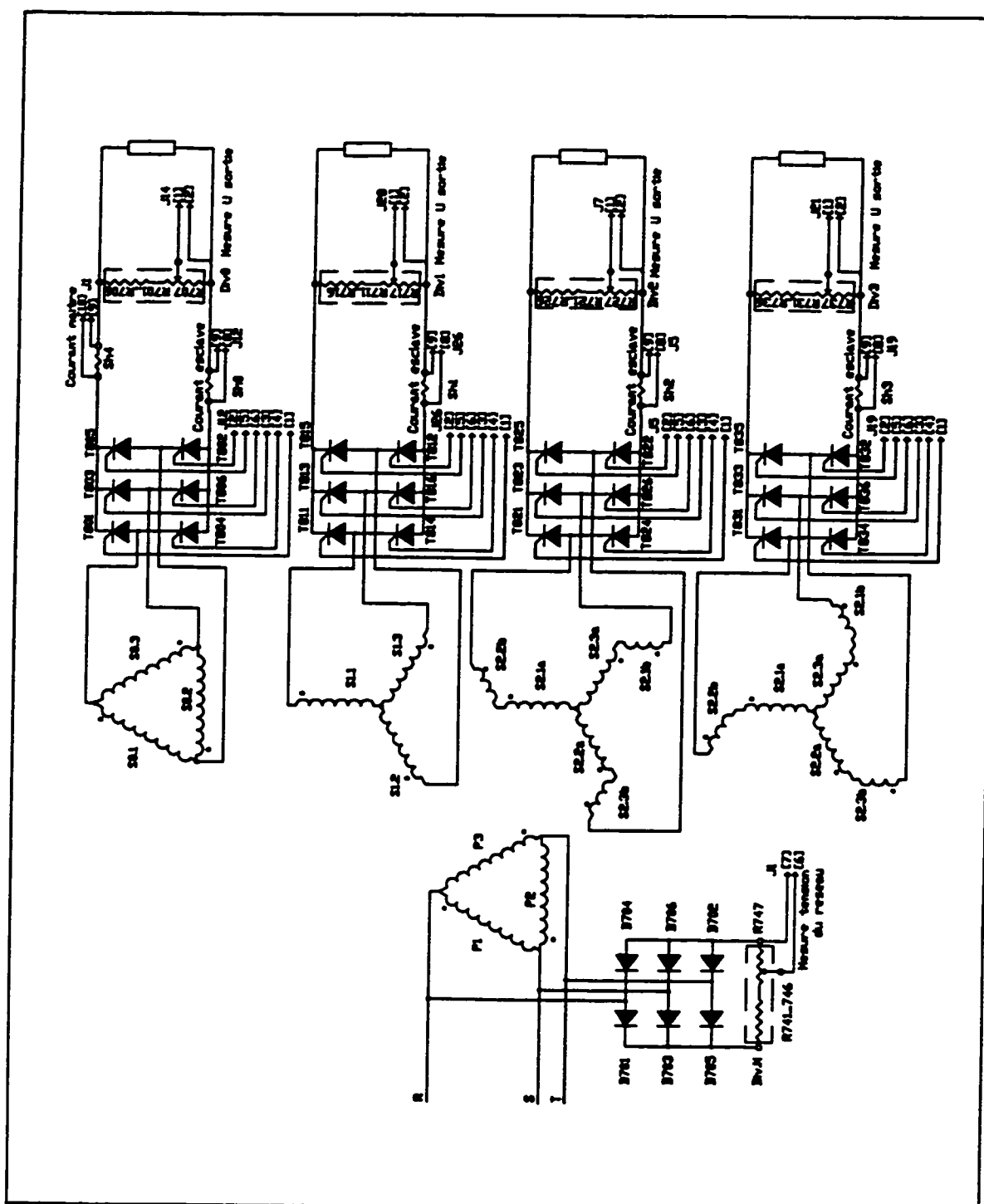


Figure 2.7 Quatre ponts de Graetz indépendants.

2.1.4 Pont triphasé hybride

Le schéma est identique à celui présenté pour le pont de Graetz (figure 2.5). Les thyristors 1, 3, 5, peuvent être commandés en utilisant la figure 2.6. Les thyristors 2, 4, 6 fonctionnent comme des diodes (ie: $\alpha = 0^\circ$).

Dans un tel cas, la tension est définie comme suit:

$$E_o = \frac{3\sqrt{2}}{\pi} E_u \left(\frac{1 + \cos \alpha}{2} \right); \quad (2.7)$$

Le courant continu maximal fourni par le redresseur est indiqué par:

$$I_{o\infty} = 3 I_{F(AV)}; \quad (2.8)$$

2.2 Redresseurs utilisant plusieurs micro-contrôleurs esclaves

L'utilisation de redresseurs à grand nombre de phases nécessite la commande de plus de six thyristors, ce qui implique l'utilisation de plusieurs micro-contrôleurs esclaves.

2.2.1 Redresseur dodécaphasé à transformateurs d'interphase

Ce convertisseur (fig. 2.8) est constitué de deux étoiles triphasées avec transformateur d'interphase, connectées entre elles par un troisième

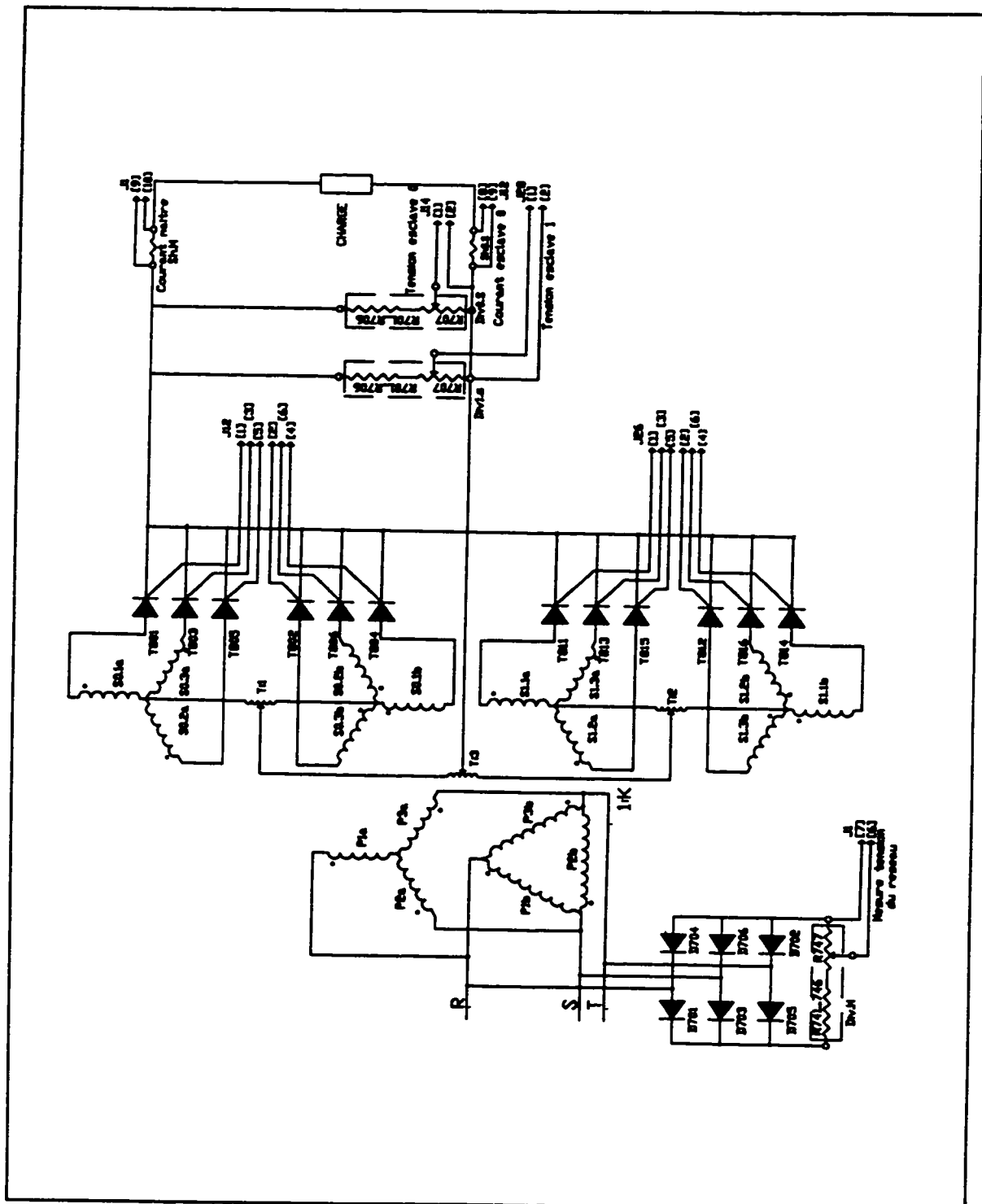


Figure 2.8 Redresseur dodécaphasé à transformateurs d'interphase.

transformateur d'interphase. La tension redressée a un indice de pulsation de douze. Les quatre groupes de tensions triphasées, fournis par les secondaires, sont déphasés de $\frac{\pi}{6}$ radians. La tension continue de sortie est définie comme suit:

$$E_o = \frac{3\sqrt{3}}{\pi\sqrt{2}} E_{ph} \cos \alpha = 1,17 E_{ph} \cos \alpha ; \quad (2.9)$$

Le courant continu de sortie du redresseur dépend du courant effectif des thyristors:

$$I_{0\infty} = 12 I_{F(AV)} ; \quad (2.10)$$

La séquence de commande de la figure 2.6 s'applique à chaque sous-ensemble formé d'une double étoile triphasée avec transformateur d'interphase (1-2, 3-4).

2.2.2 Connexion parallèle des ponts

La connexion parallèle des ponts est utilisée pour des redresseurs à courants forts qui fournissent une basse tension.

La figure 2.9 présente le schéma de la connexion parallèle de quatre ponts. La commande des thyristors générée par chaque micro-contrôleur esclave est identique à la commande d'un pont de Graetz (section 2.1.3).

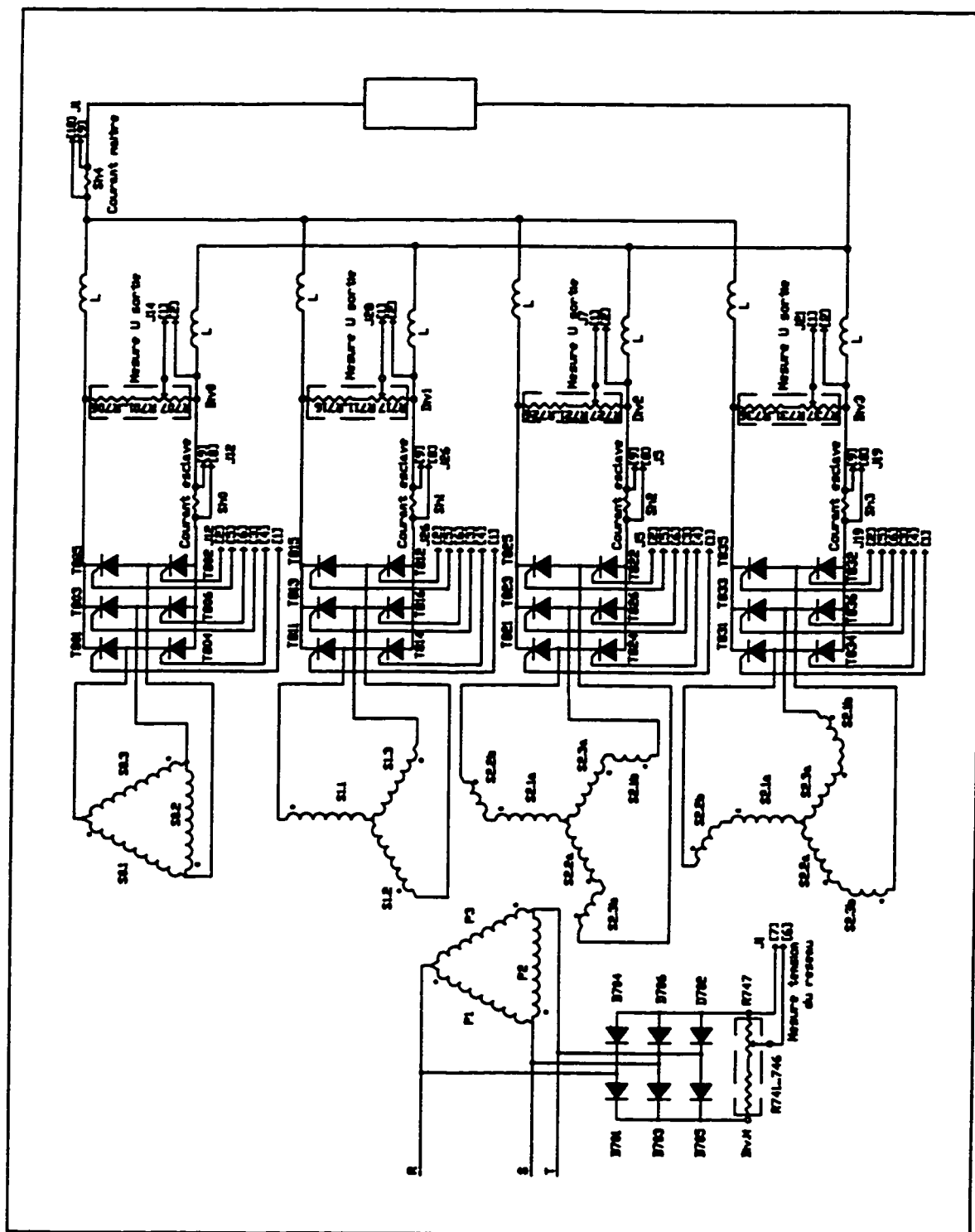


Figure 2.9 Redresseur à vingt-quatre phases - connexion parallèle.

Chaque esclave synchronise sa commande avec le système de tensions triphasées qui alimente son pont. Donc, entre les commandes des ponts, il y aura un déphasage égal au déphasage entre les systèmes de tensions qui les alimentent. Le maître devra régler le courant de chaque pont de façon égale.

La tension de sortie du redresseur a un indice de pulsation de vingt-quatre. La tension de sortie pour des ponts en parallèle se définit comme étant la tension de sortie d'un pont de Graetz:

$$E_o = E_u \sqrt{2} \frac{\sin \frac{\pi}{6}}{\frac{\pi}{6}} \cos \alpha = 1,35 E_{ph} \cos \alpha ; \quad (2.11)$$

Le courant maximal du redresseur est donné par:

$$I_{o\infty} = 12 I_{F(AV)} ; \quad (2.12)$$

Pour un redresseur dodécaphasé, on considère le sous-schéma constitué par deux ponts de Graetz. La tension de sortie aura l'indice de pulsation de douze et sera la même que celle présentée à l'équation (2.9).

Le courant maximal de sortie est indiqué par:

$$I_{o\infty} = 6 I_{F(AV)} ; \quad (2.13)$$

2.2.3 Connexion en série des ponts

La connexion en série des ponts permet d'obtenir des hautes tensions à des petits courants. La figure 2.10 montre le schéma du redresseur réalisé avec quatre ponts connectés en série. La séquence de commande des thyristors est identique à celle des ponts connectés en parallèle.

La tension du redresseur est composée des tensions de sortie des quatre ponts:

$$E_o = E_{o,0} + E_{o,1} + E_{o,2} + E_{o,3} ; \quad (2.14)$$

Où $E_{o,j}$ (j de 0 à 3) représente la contribution du pont j à la tension de sortie.

Le pont j est alimenté par les tensions de ligne $E_{ll,j}$ produites par le secondaire triphasé j . L'expression de la tension continue fournie par un pont triphasé avec diode de retour dépend de l'angle de commande α .

Pour α de 0 à $\frac{\pi}{3}$:

$$E_{o,j} = \frac{3\sqrt{2}}{\pi} E_{ll,j} \cos \alpha ; \quad (2.15)$$

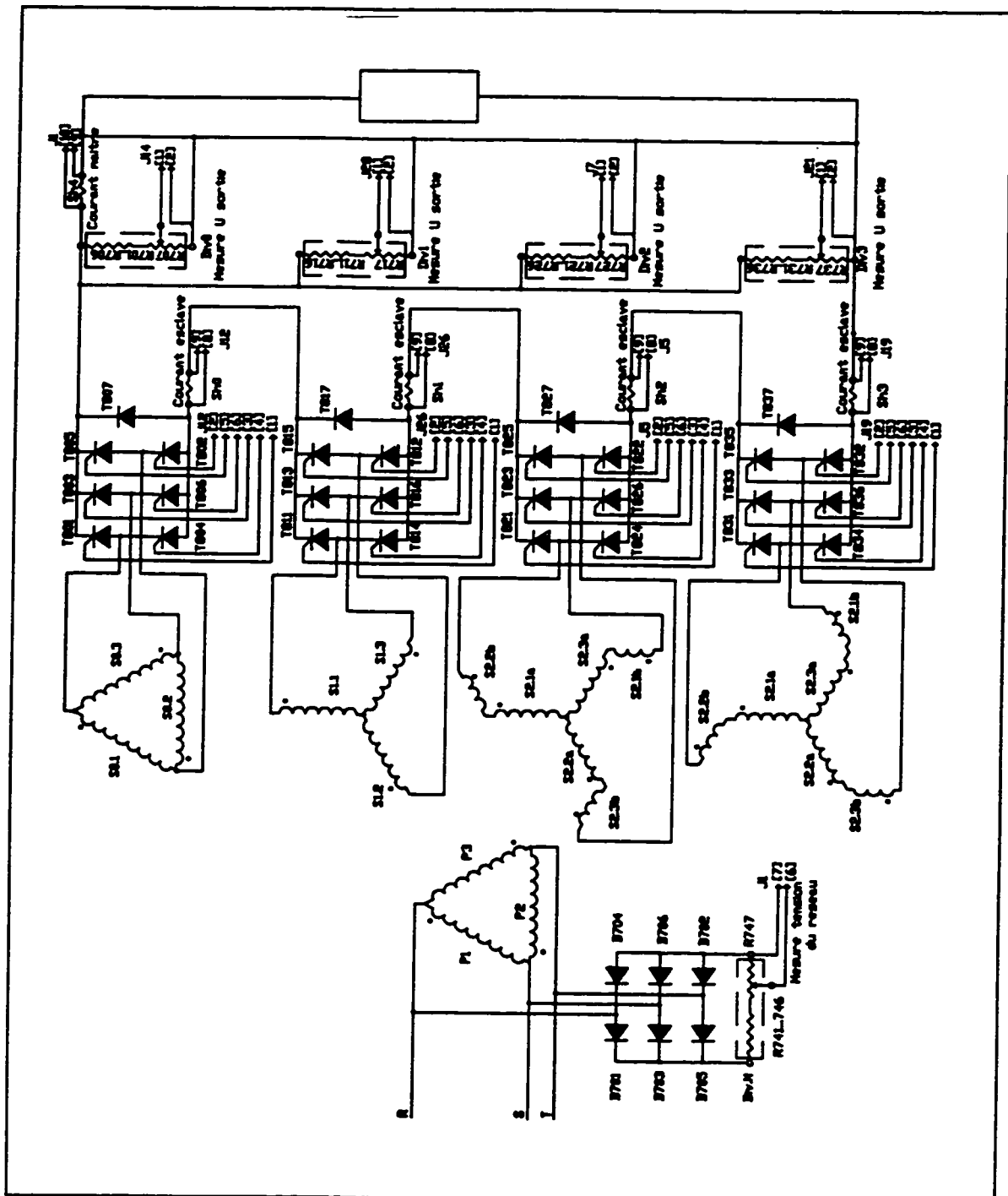


Figure 2.10 Redresseur à vingt-quatre phases - connexion série.

Pour α de $\frac{\pi}{3}$ à $\frac{2\pi}{3}$:

$$E_{o,j} = \frac{3\sqrt{2}}{\pi} E_{u,j} \left(1 + \cos\left(\frac{\pi}{3} + \alpha\right) \right); \quad (2.16)$$

On calcule le courant maximal de sortie en fonction de la capacité en courant d'un thyristor:

$$I_{o\infty} = 3 I_{F(AV)}; \quad (2.17)$$

Pour un redresseur dodécaphasé, on considère le sous-schéma constitué par deux ponts. La tension de sortie aura un indice de pulsation douze et une composante continue:

$$E_o = E_{o,0} + E_{o,1}; \quad (2.18)$$

Où $E_{o,j}$ se définit par l'équation 2.15.

Le courant de sortie est identique à celui du redresseur à 24 phases (2.17).

Toutes les configurations présentées dans ce chapitre ont plusieurs éléments communs nécessaires à leur fonctionnement:

- le shunt « sh.M » est connecté en série avec la charge. À l'aide de ce shunt le micro-contrôleur maître mesure le courant de la charge et assure une protection rapide de courant en cas de surintensité.

- chaque micro-contrôleur esclave synchronise la commande de ses thyristors avec la tension triphasée du secondaire du transformateur.
- la tension du réseau est mesurée en utilisant un pont triphasé auxiliaire D701 ... D706.
- chaque circuit de charge contient au moins un diviseur de tension pour la mesure de la tension de sortie et un shunt pour la mesure du courant de charge.

Dans le deuxième chapitre nous avons présenté les configurations pour lesquelles on a développé le système de commande. Certainement, la logique de commande des thyristors sera adaptée pour l'évaluation expérimentale des nouvelles familles de convertisseurs étudiées à la section électrotechnique de l'École Polytechnique de Montréal. Le chapitre suivant décrit une autre composante importante du redresseur commandé: le circuit d'interface entre l'étage de puissance et le micro-ordinateur qui assure la commande.

3. CIRCUIT D'INTERFACE ENTRE LE MICRO-ORDINATEUR ET L'ÉLECTRONIQUE DE PUISSANCE

Le chapitre qui suit a pour but de présenter la carte Redforth conçue dans le cadre de cette maîtrise. Cette carte constitue l'interface entre les redresseurs de l'électronique de puissance présentés au chapitre 2 et le micro-ordinateur Conforth. La carte Redforth assure la commande des thyristors, les protections du système de commande et la mesure des paramètres extérieurs tels que la tension du réseau, la tension de sortie et le courant de charge. La carte d'interface permet de commander jusqu'à vingt-quatre thyristors et de mesurer cinq tensions et cinq courants. Ce chapitre présente l'architecture du micro-ordinateur Conforth, ainsi que la carte d'interface Redforth.

3.1 Architecture du micro-ordinateur Conforth

Le système de commande des redresseurs se base sur le micro-contrôleur Conforth conçu au Département de Génie Électrique de l'École Polytechnique de Montréal [1]. Le micro-ordinateur Conforth est destiné à un usage général comme le contrôle de la circulation routière ou la gestion des caisses enregistreuses. La grande capacité de calcul et la souplesse de la structure

interne des microprocesseurs (minuterics, sources d'interruptions, convertisseurs A/N et ports de communication) rendent son utilisation intéressante dans des systèmes de commande en temps réel. Le micro-ordinateur contient la hiérarchie " maître" - quatre "esclaves" (fig. 3.1), ce qui le rend très utile dans des systèmes où plusieurs tâches doivent s'exécuter en parallèle. La commande des redresseurs à grand nombre de phases étant une des applications envisagées au moment de la conception de la carte.

Le système de commande des redresseurs se divise en deux niveaux:

- un niveau décisionnel qui impose la consigne aux micro-contrôleurs esclaves en fonction de l'ensemble des paramètres du système.
- un niveau exécutif qui assure la commande des thyristors.

Le micro-contrôleur maître, un microprocesseur Mitsubishi 37700 à 16 bits, réalise les tâches du niveau décisionnel. Il calcule les consignes et les transmet aux micro-contrôleurs esclaves. Le microprocesseur maître peut transmettre la consigne en même temps à tous les microprocesseurs esclaves ou peut envoyer à chacun une consigne particulière. En plus, le micro-contrôleur 37700 mesure la tension du réseau et le courant de sortie.

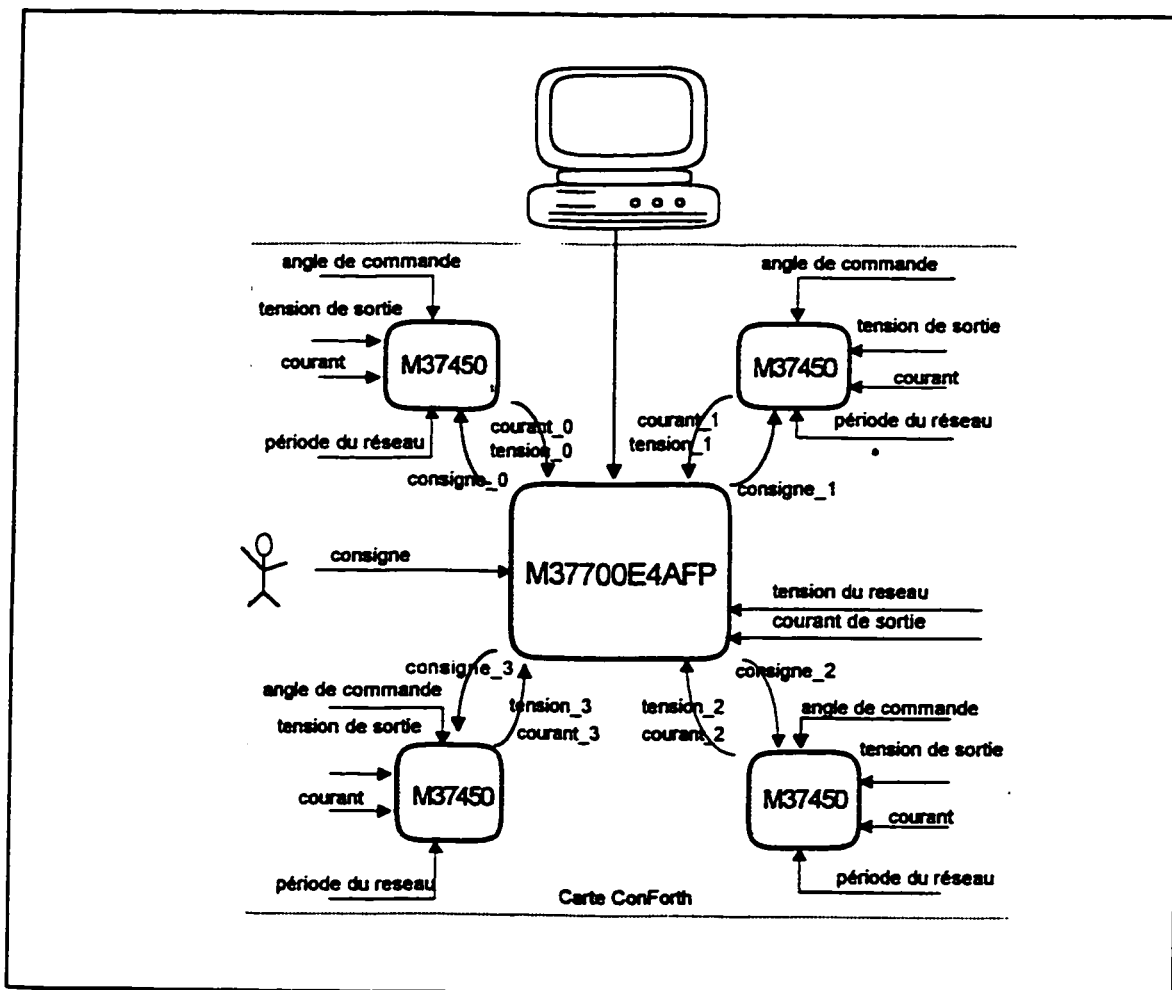


Figure 3.1 Architecture et flux de données du système.

Une connexion RS232 relie le système de commande à un ordinateur personnel. Ce lien permet la transmission des données entre le système de commande et un ordinateur situé à moins de 16 mètres. La longueur maximale du câble est spécifiée de façon indirecte par la capacité totale du câble et du récepteur qui ne doit pas dépasser 2500 pF pour assurer une transmission jusqu'à une vitesse de 20000 bits / seconde.

L'ordinateur transmet les consignes de tension au micro-ordinateur et reçoit de celui-ci des données concernant l'état du système (tensions de sortie, courants de sortie, alarmes et consignes exécutées). Les paramètres de communication sont une vitesse de transmission de 19200 bits / seconde, caractères transmis sur 8 bits, sans bit de parité et avec un seul bit d'arrêt (19200, 8N1).

Des microprocesseurs Mitsubishi 37450 à 8 bits sont utilisés comme micro-contrôleurs esclaves. Ils reçoivent la consigne du micro-contrôleur maître, calculent le nouvel angle de commande et transmettent les impulsions de gâchette appropriées aux thyristors. La transmission maître - esclaves se fait par un bus de 8 bits. Chaque micro-contrôleur esclave mesure la tension et le courant de sortie du redresseur ainsi que l'angle de commande des thyristors et la période de la tension du réseau. Chaque sous-ensemble maître - esclave est identique du point de vue matériel et logiciel.

3.2 Circuit d'interface du micro-contrôleur maître

La carte Redforth, développée au cours de cette maîtrise, réalise l'interface entre l'étage de puissance et les micro-contrôleurs maître et esclaves.

L'interface du maître isole électriquement le circuit de haute tension de celui de la commande. Simultanément, il fournit au microprocesseur les signaux analogiques: la tension du réseau (U_r), le courant de charge (I_m) et un signal

d'interruption associé à leurs variations rapides. L'interface fournit également au maître un signal d'interruption qui correspond au dépassement du courant nominal du redresseur. La valeur du courant maximal du redresseur se détermine en fonction de la configuration choisie en utilisant les équations présentées au chapitre 2.

3.2.1 Mesure de la tension du réseau

Le réseau résistif externe « div.4 » divise la tension redressée par les diodes D700...D705 (fig. 2.3 à 2.9 selon la configuration utilisée) et la tension résultante se retrouve sur le connecteur J1(4,5) (figure 3.2). La résistance variable R747 (fig. 2.3 à 2.9) sert au réglage grossier de la tension mesurée. Ce circuit permet la mesure de la moyenne des trois tensions de phase. L'absence d'une phase se traduit par une tension moyenne mesurée plus basse. Les diodes D4, D5 (fig. 3.2) limitent la tension maximale applicable au connecteur de la carte à ± 5.6 V. L'amplificateur opérationnel, U5 (AD204), transmet le signal utile et isole électriquement le circuit de puissance de celui de commande. Le potentiomètre R5 permet de régler la tension de décalage et R6 permet l'ajustement fin de l'amplitude du signal mesuré. La sortie de l'amplificateur AD204 est connectée à un amplificateur suiveur à cause

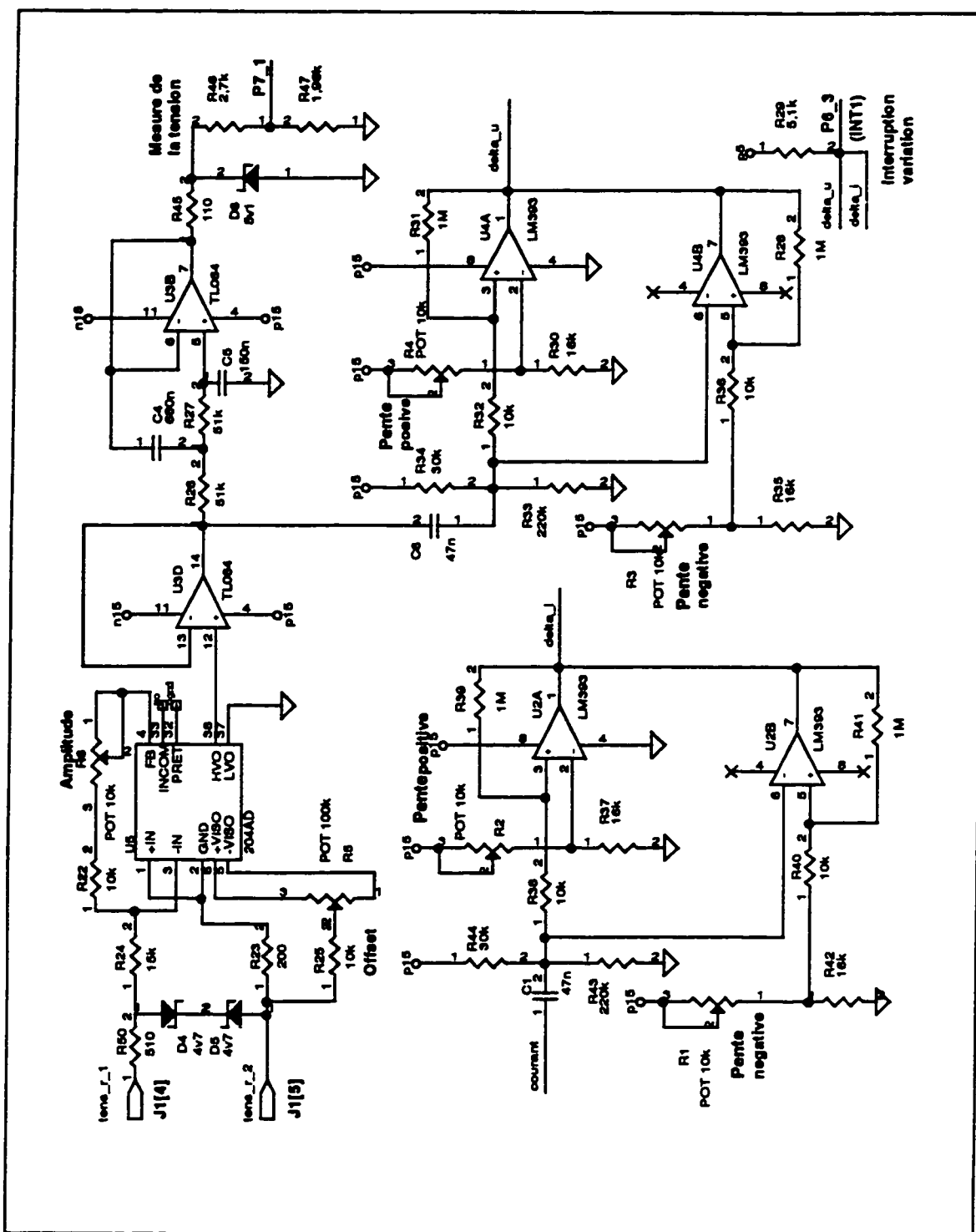


Figure 3.2 Circuit d'interface du μP maître: la mesure de la tension.

de son impédance de sortie élevée (~ 3 kilohms) qui dépend fortement de la température. Le courant faible d'entrée du suiveur minimise l'influence de la température sur le signal utile.

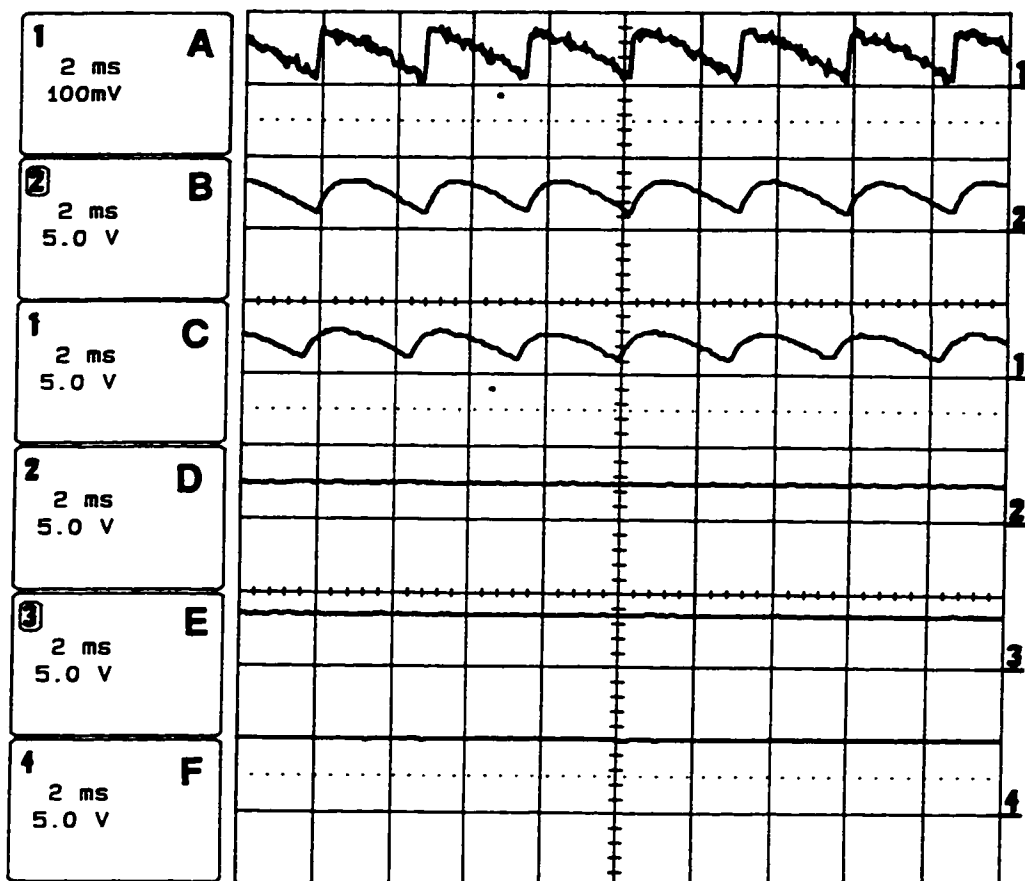
Les composants U3b, R26, C4, R27 et C5 forment un filtre du deuxième ordre passe-bas (fréquence de coupure de 60 Hz) qui élimine la composante alternative du signal de tension de réseau. Le premier harmonique de tension se situe à une fréquence de 360 Hz. La conception du filtre permet d'atténuer fortement cet harmonique et les suivants. Le signal filtré est mesuré par le micro-contrôleur maître au port P7.1.

3.2.2 Mesure du courant

Le micro-contrôleur maître mesure le courant de sortie à l'aide d'un shunt (25 A/50 mV) connecté en série avec la charge du redresseur et son logiciel assure une protection rapide de courant.

La figure 3.3 montre les principales formes d'ondes enregistrées dans le circuit de mesure de courant du micro-contrôleur maître.

La chute de tension du shunt (fig. 3.3.a) est conduite à la carte d'interface au connecteur J1(9,10) par des câbles blindés afin d'éviter les interférences induites. Le blindage est mis à la masse de mesure flottante seulement à un de ses bouts afin d'éviter l'apparition des courants de circulation. Les diodes zener



- a) La chute de tension sur le shunt.
- b) Le signal de courant amplifié à la sortie du circuit U63.
- c) Le signal de courant mesuré à la sortie de l'amplificateur d'isolation U64
- d) Le signal de courant filtré à la sortie de l'amplificateur U3c.
- e) Le signal de courant redressé et divisé (P7_0).
- f) Le signe du courant à la sortie du comparateur U6d.

NOTE: La mesure des signaux a et b nécessite l'utilisation d'un transformateur d'isolation pour l'alimentation de l'oscilloscope.

Figure 3.3 Circuit de mesure du courant: formes d'ondes.

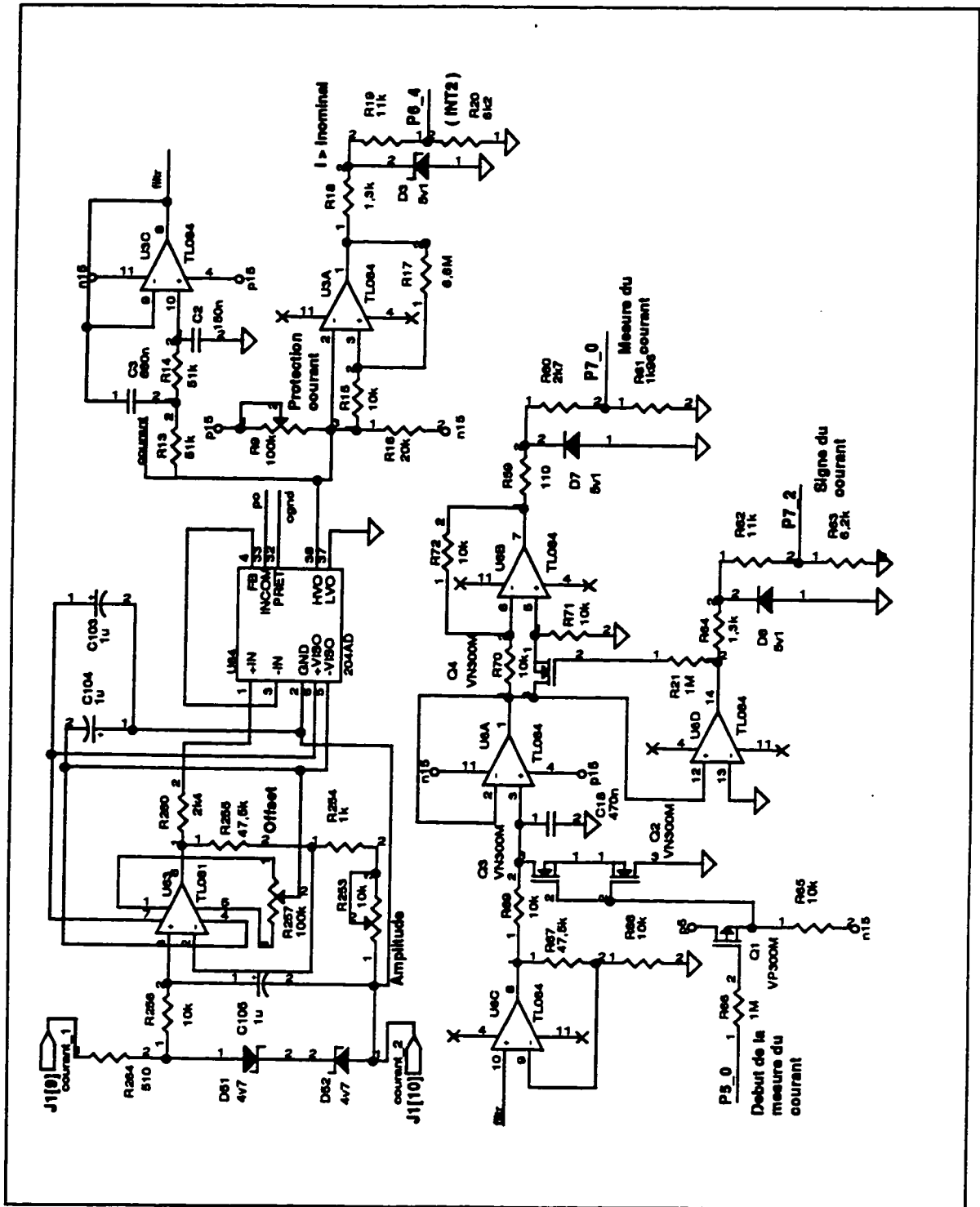


Figure 3.4 Interface du μP maître: circuit de mesure du courant.

D51 et D52 (figure 3.4) protègent l'entrée de la carte contre des tensions excédant les limites admissibles pouvant détruire les circuits U63 et U64.

Un amplificateur opérationnel, TL081 (U63), amplifie le signal de courant (gain ~ 50). Le signal amplifié est présenté à la figure 3.3.b. L'amplificateur AD204 fournit une tension bipolaire d'alimentation isolée à faible courant suffisante pour alimenter l'amplificateur opérationnel U63. Les résistances variables R257, R253 assurent le réglage de la tension de décalage et du gain de l'amplificateur. L'amplificateur U64 (AD204) réalise l'isolation électrique entre la partie de puissance et celle de commande en assurant la transmission du signal utile (fig. 3.3.c). L'alimentation de ce circuit se fait par un signal rectangulaire ayant une fréquence de 25 kHz et une amplitude de 15 V crête-crête. Le signal est produit par la minuterie TA4 du maître et est amplifié par un circuit « push-pull ». Le grand nombre de circuits AD204 (10) et le couplage capacitif à l'intérieur du AD204 nécessitent l'utilisation d'un tel montage.

Le premier harmonique du courant mesuré est de l'ordre $6n \pm 1$ (n représente le nombre de pulsations du courant de sortie pendant une période de la tension du réseau). Pour la configuration la plus défavorable, un seul pont de Graetz, n sera 1. Le premier harmonique de courant présent aura une fréquence de 360 Hz. Les composants R13, C3, R14, C2, U3c forment un filtre de deuxième ordre passe-bas qui rejette les composantes alternatives du signal mesuré (fig. 3.3.d).

Les amplificateurs U6a, U6b, U6c et U6d (fig. 3.4) amplifient et redressent sans seuil le signal de courant. Celui-ci est mesuré par le microprocesseur au port P7.0 (fig. 3.3.e). En même temps, le microprocesseur lit au port P7.2 la polarité du courant établie par le comparateur U6d (fig. 3.3.f). Le diviseur R59, R60, R61 et la diode zener D7 protègent l'entrée du microprocesseur des tensions négatives. Toute tension négative qui pourrait apparaître à la suite d'une défectuosité des composantes sera limitée à -0.6 V par D7 et divisée par R60 et R61; la tension négative maximale admissible par le μ P à chacune de ses entrées est de -0.3 V.

Le filtre RC (R69 et C18) et le commutateur statique bi-alternance (Q1, Q2, Q3), commandé par le microprocesseur (port P5.0), permettent de mesurer la valeur moyenne des échantillons du courant de sortie qui seront utiles au développement des nouvelles configurations de redresseurs.

Le tableau 3.1 présente la totalité des signaux disponibles au connecteur du micro-contrôleur maître. La carte Redforth n'utilise ni l'alimentation ni la masse de la carte Conforth. Les masses des deux cartes ont un seul point commun, situé à la borne de masse de la source commune de $\pm 15V$ de façon à limiter la propagation du bruit des circuits numériques vers les circuits analogiques.

Tableau 3.1: Signaux du connecteur d'entrées / sorties du μ P M37700.

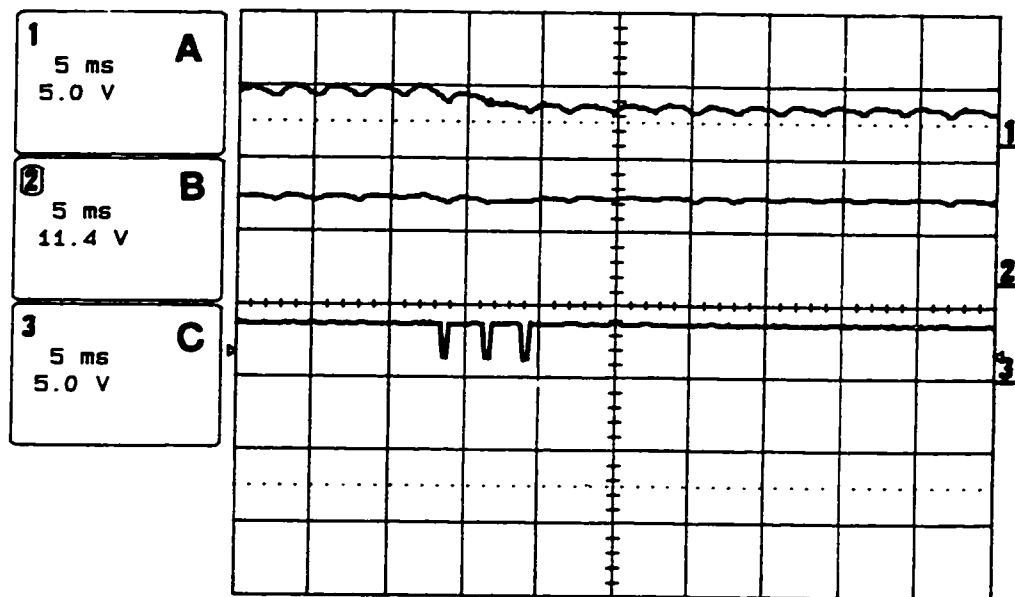
Broche	Nom	Utilisation
1	Vin	Alimentation avant régulation.
2	+5v	Alimentation régulée.
24 26 28 30 32 34 36 38 40	Gnd	Masses numérique et analogique. (pas utilisées)
4	RST*	Remise à zéro du 37700.
6	CLR*	Remise à zéro de la carte (sortie seulement)
5	PHI	Horloge du 37700.
7	E*	« Enable » du 37700 amplifié et retardé.
39	Vref	Référence du convertisseur A/N (selon W1).
23	P7.0	Mesure du courant.
25	P7.1	Mesure de la tension du réseau.
27	P7.2	Signe du courant mesuré.
16	P6.0	Source d'alimentation des AD204 (15 kHz).
3	P6.2 / INT0	Interruption des co-processeurs.
18	P6.3 / INT1	Interruption pente dépassée (courant ou tension).
19	P6.4 / INT2	Interruption de dépassement du courant nominal.
8	P5.0	Commande de la mesure de tension.
9	P5.1	
10	P5.2 / D9	Diode verte: Le système exécute le réglage.
11	P5.3 / D10	Diode rouge: Arrêt d'urgence de tous les esclaves.
12	P5.4 / D11	Diode rouge: Protection de courant du maître
13	P5.5 / D12	Diode rouge: Protection de courant des esclaves.
14	P5.6 / D13	Diode jaune: Erreur de réglage.
15	P5.7 / D14	Diode rouge: Protection de tension du maître

3.2.3 Détection de la variation des conditions extérieures

Les variations rapides du courant ou de la tension du réseau produisent une demande d'interruption pour le μ P maître qui conduira à la mise à jour des variables du système. En plus, un courant de charge qui dépasse une valeur prédéterminée entraîne un autre type de requête d'interruption au micro-contrôleur maître qui va déclencher la protection rapide de courant. Le circuit U3a compare le signal de courant à la sortie de U64 (figure 3.4) au courant maximal du redresseur (déterminé par le potentiomètre R9). Le résultat de la comparaison constitue le signal d'interruption INT2 pour le microprocesseur maître.

La figure 3.5 présente la variation décroissante du courant de sortie (fig. 3.5.a) qui génère des demandes d'interruption (fig. 3.5.c). Le signal de courant est dérivé (fig. 3.5.b) par C1, R44, R43 et ensuite appliqué aux comparateurs U2a et U2b (fig. 3.2). Ce circuit compare la pente du signal à celles déterminées par les potentiomètres R1 (pente négative) et R2 (pente positive). Une pente trop grande (en valeur absolue) met à zéro la sortie du comparateur. Les sorties des comparateurs forment un circuit « ou » logique. Leur passage à zéro détermine une interruption INT1 au niveau du micro-contrôleur maître.

Un circuit similaire à celui qui surveille la pente du courant surveille la pente de la tension (fig. 3.2).



- a) Variation du signal de courant de charge à la sortie de l'amplificateur U64.
- b) Signal de courant dérivé à l'entrée de U2a.
- c) Demandes des interruptions transmises au maître (P6.3).

Figure 3.5 Détection de la variation du courant de sortie: formes d'ondes.

3.2.4 Affichage lumineux de l'état du système (alarmes)

Six diodes lumineuses (fig. 3.6) affichent l'état du système et signalent les alarmes:

- La diode D9 (verte) s'allume si le système exécute le réglage. La diode s'éteint si le système est arrêté.
- La diode D10 (rouge) s'allume si le maître annule toutes les commandes des thyristors (port P5.3 mis à "1"). Cette protection permet d'arrêter les thyristors si le maître devient inactif suite à une erreur du programme ou à une perte de l'alimentation. La mise en route du système annule toutes les commandes des thyristors.
- La diode D11 (rouge) clignote si le courant du maître est anormal. Elle reste allumée continuellement si le système est arrêté à cause d'une surcharge de longue durée détectée par le maître.
- La diode D12 (rouge) clignote si le courant des esclaves est anormal. La diode reste allumée continuellement si le système est arrêté à cause d'une surcharge de longue durée détectée par les esclaves.
- La diode D13 (jaune) s'allume si une erreur de réglage est détectée.

- La diode D14 (rouge) clignote une fois si la tension du réseau est trop basse, ou deux fois si la tension est trop haute.

3.3 Circuit d'interface du microprocesseur esclave

Le μ P 37450 a deux modes de fonctionnement possibles: le mode autonome et le mode esclave (dépendant d'un microprocesseur maître). La carte Conforth utilise cette dernière configuration.

L'esclave doit commander un pont de Graetz, recevoir la consigne d'angle, mesurer et transmettre de nouvelles données au micro-contrôleur maître tel que la tension et le courant de sortie.

L'interface de l'esclave contient un circuit numérique destiné à la commande des thyristors et une partie analogique utilisée pour l'acquisition des données (tension et courant de sortie).

Les paragraphes qui suivent présentent les schémas du pont zéro sur la carte Redforth. Les circuits des trois autres ponts sont identiques. Nous appelons le pont zéro le pont commandé par le micro-contrôleur zéro.

3.3.1 La mesure de la tension de sortie

Le circuit de mesure de la tension de sortie des micro-contrôleurs esclaves (figure 3.7) contient des éléments similaires à celui de mesure de la tension du réseau et les principales formes d'ondes sont montrées à la figure 3.8. Le circuit résistif « div.0 » (fig. 2.3 à 2.9), divise la tension de sortie du pont. Le potentiomètre R707 de « div.0 » permet un ajustement grossier de la tension mesurée. La tension réduite du diviseur (fig. 3.8.a) est appliquée au connecteur J7(1-2). Les diodes D17 et D16 limitent la tension appliquée pour protéger l'amplificateur opérationnel U8 contre les tensions excessives. R74 permet l'ajustement de la tension de décalage et R75 permet l'ajustement fin de la mesure de tension (fig. 3.7).

Le signal de mesure est échantillonné à l'aide d'un commutateur bidirectionnel (Q7, Q8, Q9) commandé par le μ P esclave (fig. 3.8.c) au port P6.0. Un filtre RC (R103, C27) fait la moyenne du signal échantillonné (fig. 3.8.b). La mesure de la tension de sortie par échantillonnage est utile pour des topologies de ponts en parallèle ou série afin de déterminer la contribution de chaque pont à la tension de sortie. Les circuits U20a, b, c, d forment un redresseur de précision. Les résistances R109, R108 et R104 doivent être de bonne précision (1%) afin d'obtenir une tension redressée correcte. Le μ P mesure la tension redressée (fig. 3.8.d) au port analogique P4.0 et il lit au port P4.2 la polarité du signal.

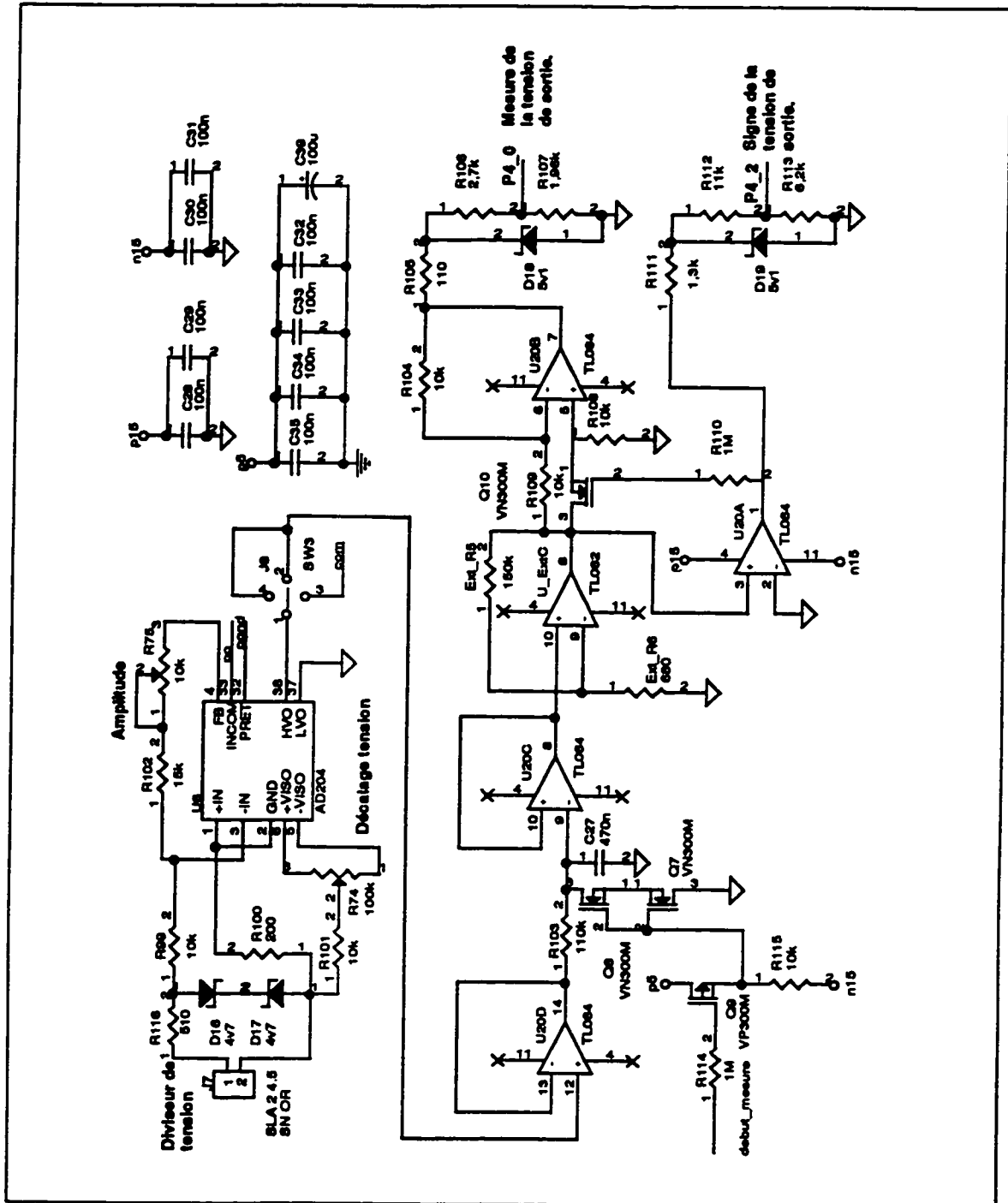
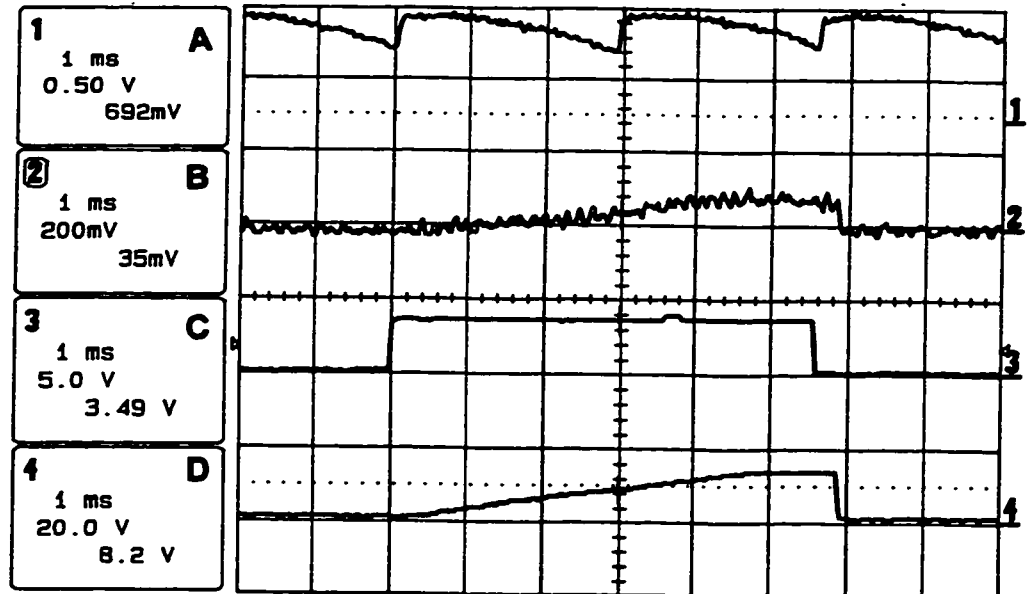


Figure 3.7. Interface du µP esclave: circuit de mesure de la tension de sortie du redresseur.



- a) La tension redressée mesurée à la sortie de l'amplificateur U8.
- b) L'échantillon de tension intégré aux bornes du condensateur C27.
- c) La commande de l'échantillonneur (la grille de Q9).
- d) La tension redressée mesurée par le μP au port P4.0. La mesure de la tension se fait à la moitié de l'intervalle d'intégration.

Figure 3.8 Mesure de la tension de sortie: formes d'ondes.

3.3.2 Mesure du courant de sortie

Le schéma électrique des esclaves est similaire à celui du micro-contrôleur maître. R117 et C22 forment le premier pôle du filtre, auquel s'ajoutent deux autres pôles (R83, C26, R86, C25) afin d'éliminer les composantes alternatives du signal mesuré (fig. 3.9). Le μ P esclave mesure le courant à l'entrée analogique P4.1 et lit sa polarité au port P4.4 afin de déterminer le sens du courant.

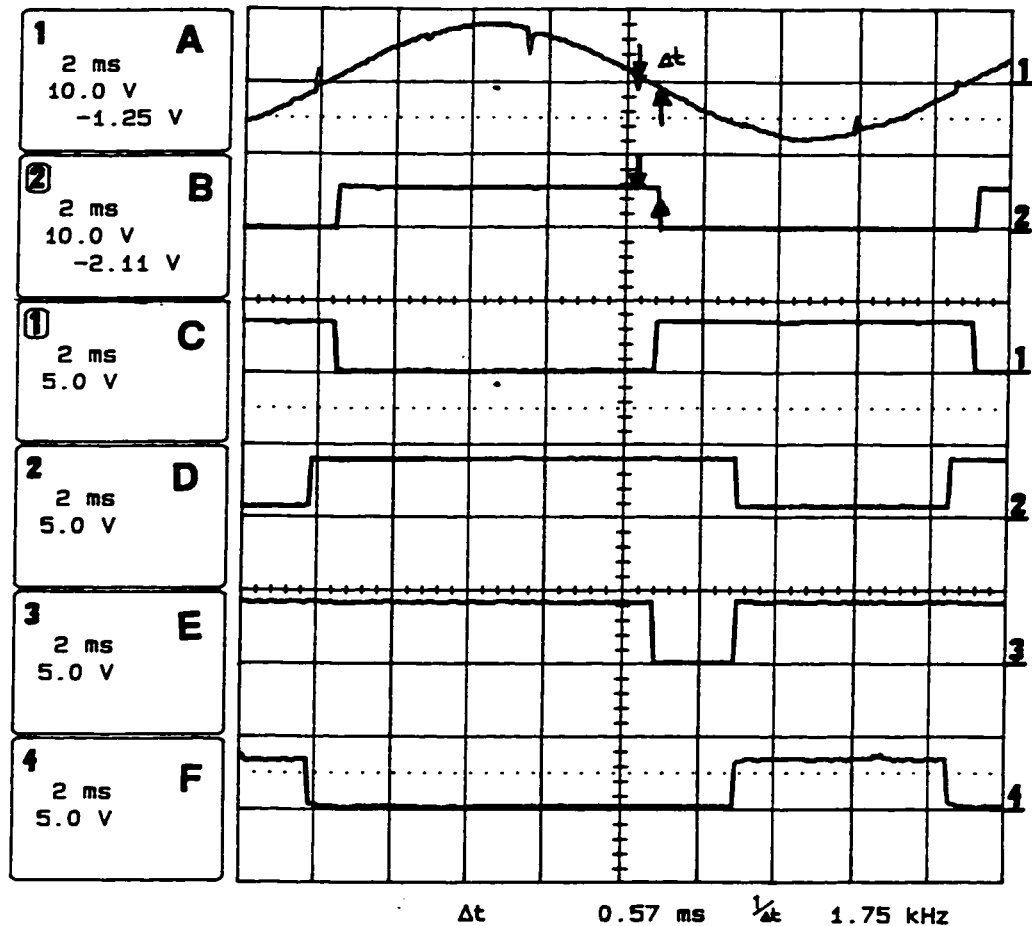
3.3.3 Commande des thyristors

Les thyristors sont mis en conduction en utilisant la méthode de commande équidistante [4]. La succession des commandes des thyristors est périodique pour un régime établi et elle sera présentée au quatrième chapitre. La commande des thyristors est synchronisée à la tension triphasée qui alimente le redresseur à l'aide des cartes de détection des passages par zéro CATFIL. Le μ P esclave calcule le délai de temps entre les commandes envoyées aux thyristors en fonction de la consigne d'angle. Par définition, l'angle de commande d'un thyristor d'un pont triphasé représente l'angle mesuré entre le point de « commutation naturelle en diode » du thyristor et le moment où le même thyristor reçoit l'impulsion de gâchette.

Figure 3.9 Interface du μP esclave: circuit de mesure du courant de sortie.

Pour un pont de Graetz, le point de « commutation naturelle en diode » correspond au passage par zéro de la tension de ligne U_{CA} (fig. 3.10.a). Le micro-contrôleur esclave mesure l'angle de commande seulement une fois par période, lors de la commande du thyristor 1.

Les cartes CATFIL reçoivent un signal proportionnel à la tension de ligne U_{AC} (fig. 3.10a) et elle fournissent un signal (fig. 3.10.b) qui indique les passages par zéro de la tension U_{CA} . Ces cartes contiennent un filtre passe-bas de deuxième ordre, qui rejette les fréquences plus élevées que la fondamentale, et un comparateur de précision. Le filtre introduit un déphasage qui est compensé par le logiciel et qui limite l'angle minimal de commande des thyristors. Le délai introduit par le filtre est indiqué et mesuré sur les formes d'onde de la figure 3.10. Le signal des passages par zéro est branché à la carte Redforth au connecteur J6(1,2) (fig. 3.12). L'intervalle entre deux passages par zéro de la tension U_{CA} correspond à la période de la tension du réseau mesurée par le μP au port P3.1 (fig. 3.10.c). Le circuit U11a (fig. 3.13) fournit une impulsion (fig. 3.10.e) qui débute au passage par zéro de la tension U_{CA} (fig. 3.10.c) et qui finit au moment où la commande du premier thyristor commence (fig. 3.10.d). La durée de cette impulsion représente l'angle de commande du thyristor 1 qui est mesuré par le μP au port P3.2.



- a) La tension de ligne U_{CA} mesurée à l'entrée de la carte CATFIL.
- b) Le signal obtenu à la sortie de la carte CATFIL.
- c) Signal utilisé pour la mesure de la période de la tension (P3.1).
- d) La commande du thyristor 1 mesurée au connecteur J5-1 (thyristor en conduction pour le niveau "0" du signal).
- d) Signal utilisé pour la mesure de l'angle de commande de thyristor 1 (P3.2). La mesure est faite pour le niveau "0".
- e) Le signal utilisé à la commande de l'échantillonnage de la tension de sortie (U11b-9).

Figure 3.10 Mesure de l'angle de commande: formes d'ondes.

Tableau 3.2 Signaux du connecteur d'entrées / sorties du μ P M37450

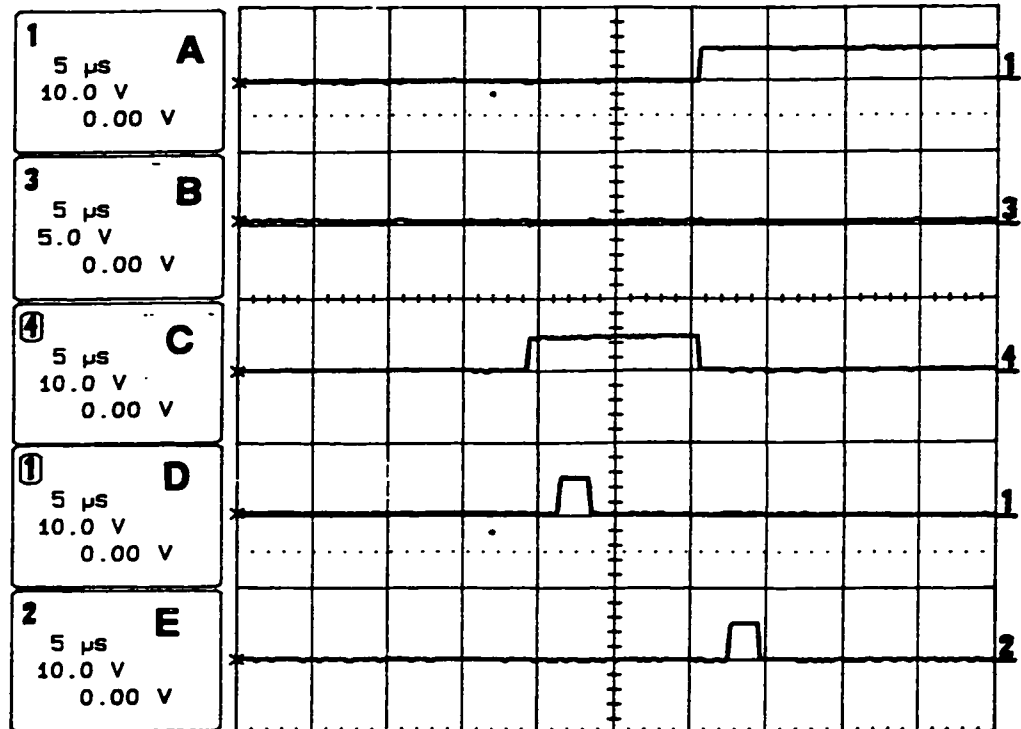
Broche	Nom	
1	Vin	Alimentation avant régulation.
2	+5v	Alimentation régulée.
24 26 28 30 32 34	Gnd	Masses numérique et analogique.
17	RST*	Remise à zéro du 37450.
4	PHI	Horloge du 37450 (cristal divisé par deux).
32	ADRef	Référence du convertisseur A/N.
33	DARef	Référence du convertisseur N/A.
20	DA1	Convertisseur numérique - analogique.
21	DA2	Convertisseur numérique - analogique.
23	P4.0	Mesure du courant.
25	P4.1	Mesure de la tension.
27	P4.2	Signe du courant mesuré.
29	P4.3	Signe de la tension mesurée.
5	P6.0	Fin de la mesure de tension
6	P6.1 / INT1	
7	P6.2 / INT2	Début de la mesure de la tension
8	P6.3	
10	P3.1	Mesure de la période de la tension du réseau.
11	P3.2	Mesure de l'angle de commande.
9	P3.0	Envoi des commandes aux thyristors.
12	P3.3	Chargement du registre des thyristors 1-3
13	P3.4	Chargement du registre des thyristors 4-6
14	P3.5	Commande des thyristors 1 et 4.
15	P3.6	Commande des thyristors 2 et 5.
16	P3.7	Commande des thyristors 3 et 6.

La commande du thyristor 1 initie également le début de l'échantillonnage (U11b) de la tension mesurée (fig. 3.10.f). La fin de l'échantillonnage est commandée par la sortie P6.0 du μ P. Si on choisit la mesure de la tension sans échantillonnage de la tension, il suffit de garder toujours la sortie P6.0 à « un » logique.

La commande des thyristors se fait en deux étapes. Lors de la première étape le mot de commande (6 bits) est choisi en fonction de l'itération courante du programme et ensuite il est envoyé aux registres externes (U14 et U15) par tranches de 3 bits au ports: P3_5, P3_6, P3_7 (fig. 3.12a, b, c). Les bits P3_3, P3_4 (fig. 3.12.d, e) servent à sélectionner le registre externe approprié (fig. 3.11) qui va enregistrer une moitié du mot de commande. Les registres enregistrent le mot de commande au front tombant du signal de l'horloge.

- Ensuite, la valeur du délai de temps qui suit l'envoi de la commande courante est calculée et envoyée au registre de la minuterie TA4. Ce délai de temps sera mesuré à la fin du décompte en cours de la minuterie.

Pendant la deuxième étape, la fin du décompte de la minuterie génère un changement d'état du port P3.0. Cette transition déclenche la transmission du mot de commande contenu dans registres externes U14 et U15, aux thyristors. De cette façon l'envoi de la commande est déterminé par le matériel et non par le logiciel. Ainsi, on évite un possible retard de la commande des thyristors



- a) La succession des deux parties du mot de commande au port P3.5.
- b) La succession des deux parties du mot de commande au port P3.6.
- c) La succession des deux parties du mot de commande au port P3.7.
- d) Le signal du port P3.3 qui a son front tombant charge l'état des ports P3.5, P3.6, P3.7 (0, 0, 1) dans le registre U14.
- e) Le signal du port P3.4 qui a son front tombant charge l'état des ports P3.5, P3.6, P3.7 (1, 0, 0) dans le registre U15.

Figure 3.11 Commande des thyristors: formes d'ondes.

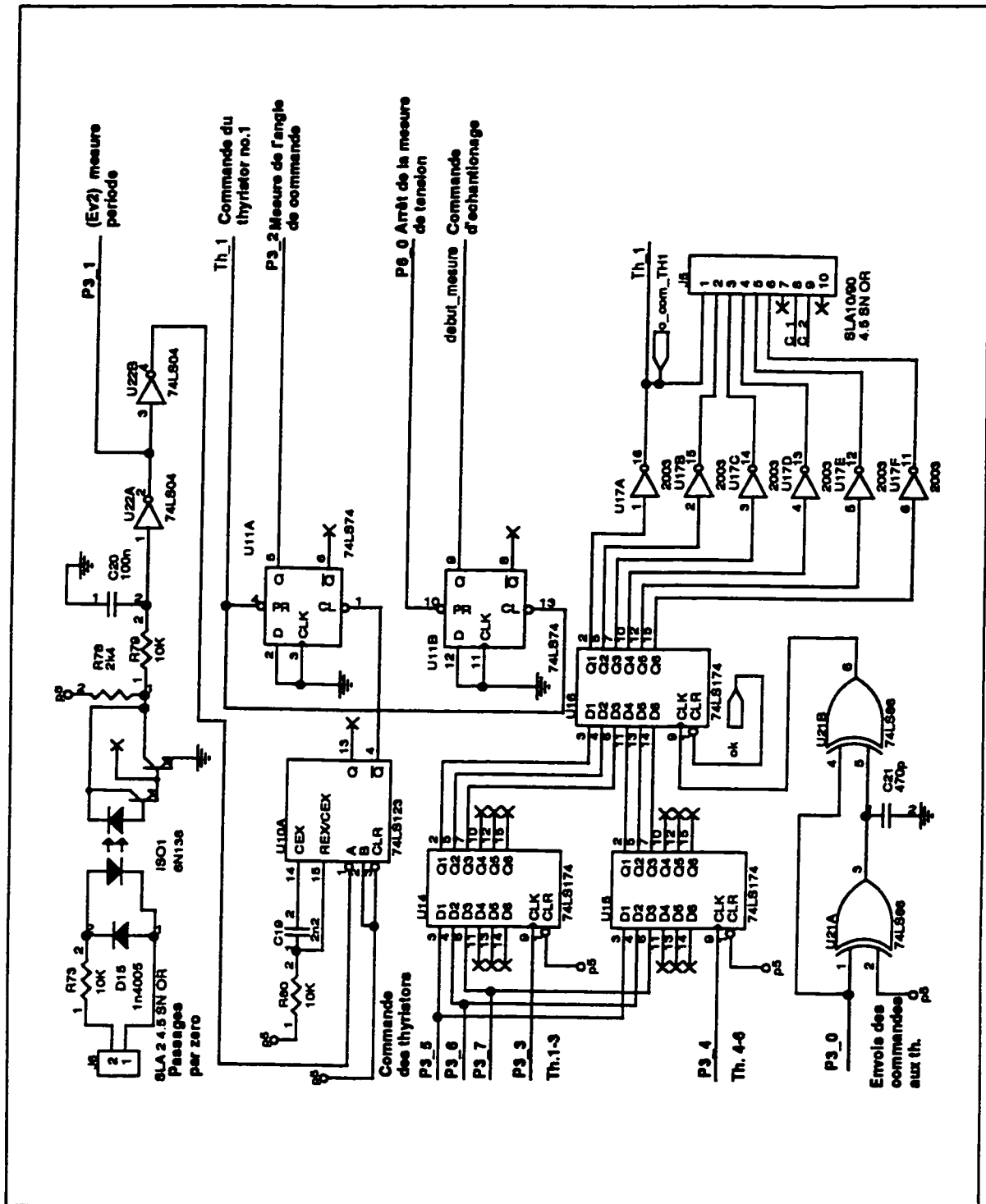


Figure 3.12 Interface esclave: circuit de commande des thyristors.

suite à l'apparition d'une interruption plus prioritaire comme celle de réception d'une consigne du micro-contrôleur maître.

En résumé la carte d'interface Redforth permet, en fonction du logiciel du maître, la commande des redresseurs à 12 et 24 phases, ainsi que quatre ponts de Graetz indépendants. Ceci est possible car chaque esclave mesure la tension et le courant du redresseur qu'il commande. Les connexions avec l'électronique de puissance sont indiquées sur les schémas électriques de ce chapitre et les connexions avec les ports du micro-contrôleur sont présentées dans les tableaux 3.1 et 3.2. Dans le chapitre suivant, nous allons décrire le logiciel du micro-ordinateur qui commande le redresseur à l'aide de la carte d'interface.

4. LOGICIEL DE COMMANDE

Le quatrième chapitre a pour objet de présenter les logiciels écrits en Forth et Assembleur pour les microprocesseurs du micro-ordinateur Conforth. La carte Conforth est composée de micro-contrôleurs maître et esclaves qui ont chacun une version du compilateur Forth [5], [6]. Le logiciel conçu pour le micro-ordinateur commande des thyristors, assure les boucles de réglage d'angle et de tension, les protections du redresseur et la communication entre les éléments du système. La totalité des sous-programmes du logiciel s'exécutent en utilisant les interruptions des microprocesseurs. Les microprocesseurs utilisés se caractérisent par un grand choix de minuteries et interruptions programmables ce qui leur donne une grande adaptabilité aux diverses applications.

Le logiciel des esclaves mesure la tension et le courant de sortie ainsi que l'angle de commande des thyristors et la période de la tension du réseau. Ces deux dernières mesures permettent au programme du micro-contrôleur esclave de réaliser une boucle de réglage de l'angle de commande des thyristors.

Le logiciel du maître calcule la consigne d'angle des micro-contrôleurs esclaves en fonction des tensions de sortie mesurées et des consignes reçues par le lien de communication RS-232. Tous ces éléments constituent la boucle de réglage de tension qui surveille la tension redressée. Simultanément, le micro-

contrôleur maître assure les protections de base d'un redresseur telles que: surcharge de courant, protection rapide de courant et tension de réseau en dehors des limites normales.

4.1 Logiciel du micro-contrôleur maître

Le programme du micro-contrôleur maître a pour objet d'assurer le contrôle de la tension de sortie en utilisant une boucle de réglage de tension. Les variables qui caractérisent le système de commande sont transmises périodiquement à un ordinateur personnel, par un lien de communication RS-232, qui les affiche à l'écran.

La figure 4.1 présente les dépendances entre les principaux sous-programmes du logiciel du micro-contrôleur maître. Ainsi, la direction des flèches montre le sens des interactions et les boucles continues adjointes aux sous-programmes indiquent l'exécution périodique de ceux-ci. Par contre, une boucle discontinue signifie une exécution périodique seulement à la suite de conditions spécifiques.

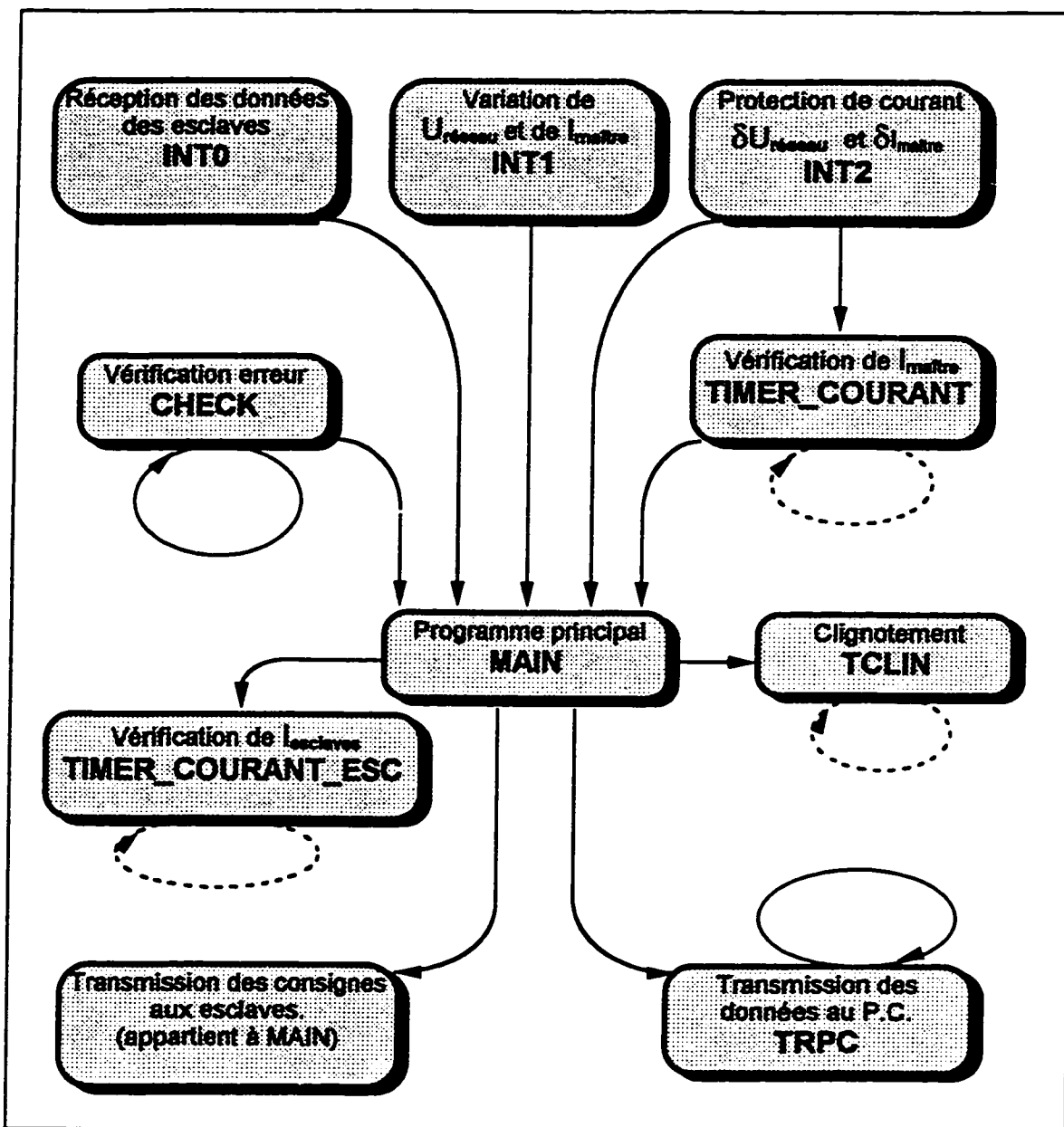


Figure 4.1 Architecture du programme du micro-contrôleur maître.

4.1.1 Sous-programmes de communication

4.1.1.1 Réception des consignes de l'ordinateur personnel

La réception des consignes par le lien de communication RS-232 se fait dans l'arrière plan sur le plus bas niveau de priorité. Cela signifie que la donnée présente au port de communication sera lue après que tous les programmes d'interruption qui ont une priorité plus haute soient exécutés.

4.1.1.2 Transmission des données à l'ordinateur personnel

Le microprocesseur maître exécute le sous-programme « *trpc* » à la suite des interruptions de la minuterie TA0. Ce programme transmet les variables d'état du système à l'ordinateur personnel et son temps d'exécution est d'un ordre de grandeur supérieur au temps d'exécution des sous-programmes de réglage. Afin de minimiser le temps d'exécution des autres programmes plus importants pour le réglage de la tension de sortie, nous avons accordé à cette interruption le plus bas niveau de priorité.

La structure symétrique du micro-ordinateur présentée au paragraphe 3.1 se reflète dans l'organisation des variables d'état (annexe A). Ainsi, les variables réelles sont enregistrées dans la matrice « état » (figure 4.2) qui contient des variables qui forment des groupes de données (tensions de sortie, courants de sortie, ...). Les variables « *Off_...* » donnent la position des groupes de données

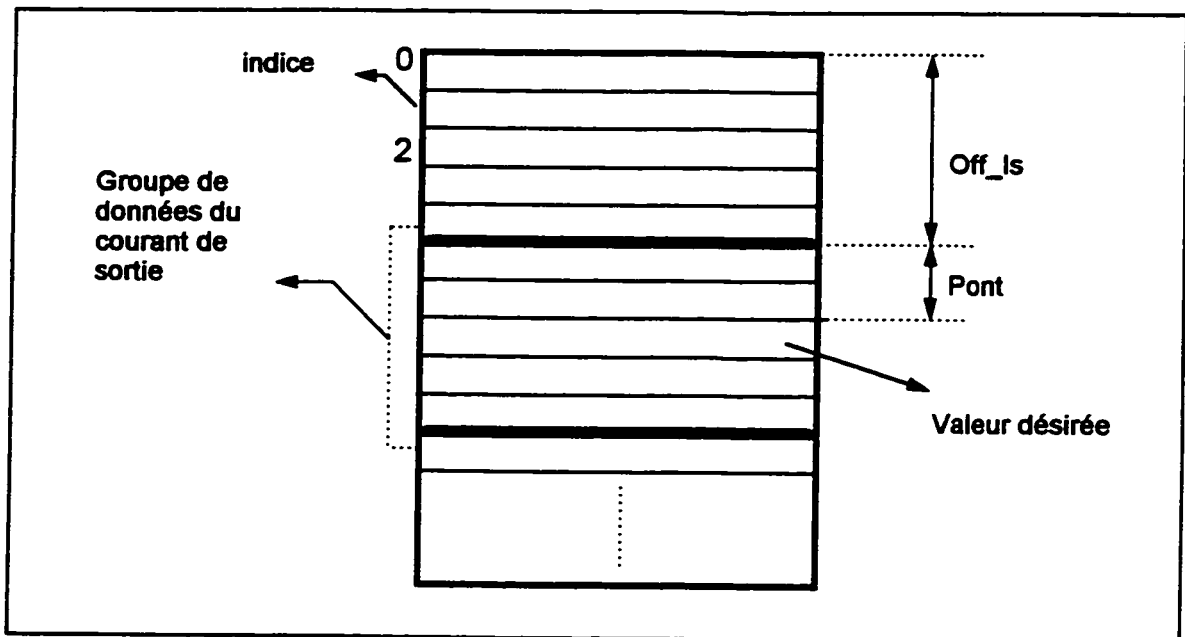


Figure 4.2 Structure de la matrice "état".

tandis que la variable « *pont* » donne la position d'une variable à l'intérieur d'un groupe de données. La variable « *pont* » est initialisée au début du programme principal avec le numéro du pont pour lequel on fait le réglage. Les valeurs 0 à 3 correspondent aux ponts 0 à 3 et la valeur 4 correspond au fonctionnement en pont équivalent (ponts en série ou parallèle qui fonctionnent avec une commande commune). Par exemple, pour accéder à la variable « *courant de sortie du pont deux* », il suffit d'ajouter la variable « *pont* » à la variable « *Off_Is* ». Le résultat représente l'indice de la matrice « état » qui pointe la donnée désirée.

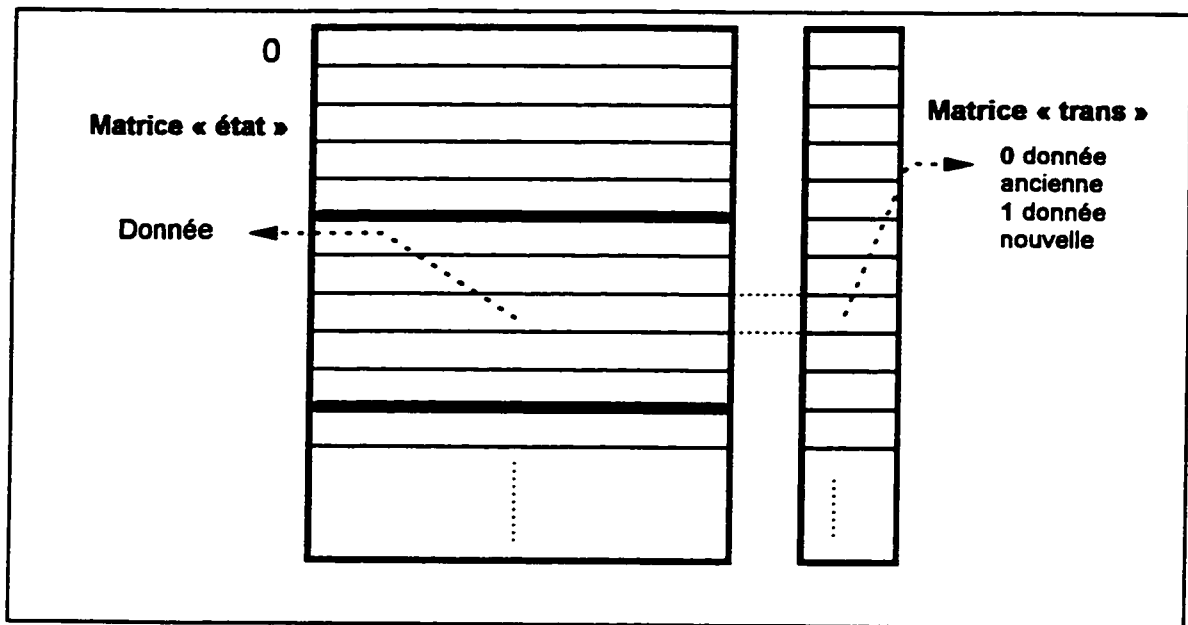


Figure 4.3 Identification des données à transmettre.

Un drapeau, situé dans la matrice « *trans* » à la même position que la donnée de la matrice état, indique si la donnée a été transmise (figure 4.3). La valeur 1 du drapeau signale la présence d'une nouvelle donnée.

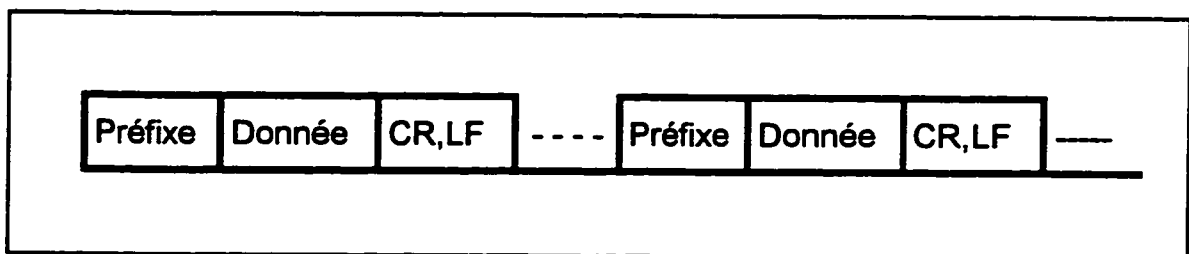


Figure 4.4: Configuration d'une trame de données.

L'interruption de la minuterie TA0 transmet périodiquement à l'ordinateur une trame de données (figure 4.4). Cette trame se compose de toutes les données

de la matrice « état » qui ont eu leur drapeau, enregistré dans « trans », égal a un. Un groupe « CR, LF » (« \r\n ») délimite la fin de chaque donnée. L'indice de la matrice (la position de la donnée dans la matrice) est transmis comme préfixe pour chaque donnée de façon à ce que l'ordinateur personnel puisse l'identifier.

La matrice d'entiers « b_état » contient une partie des variables entières sur 8 bits (annexe B) et elle a essentiellement la même structure que la matrice « état ». Nous avons défini de nouveaux mots Forth (annexe C) qui facilitent la lecture et l'écriture des données des matrices « état » et « b_état ».

4.1.1.3 Réception des données des esclaves

L'interruption INT0 représente la réaction du maître aux requêtes des esclaves qui désirent transmettre les données mesurées. La carte Conforth réalise un "et" logique entre toutes les requêtes des esclaves et le résultat génère la demande d'interruption INT0 du microprocesseur. Les demandes d'interruption sont enregistrées dans un registre de 16 bits, situé à l'adresse 01.F000H, par la mise à 1 du bit correspondant (figure 4.5). Ainsi, les bits 8 à 11 indiquent les micro-contrôleurs qui ont fait une demande d'interruption. Le sous-programme lancé par INT0 vérifie le registre à partir du bit le moins significatif et enregistre les demandes d'interruptions dans la variable « req ». De cette façon INT0

assigne aux esclaves une priorité décroissante à partir de l'esclave 0 à l'esclave 3.

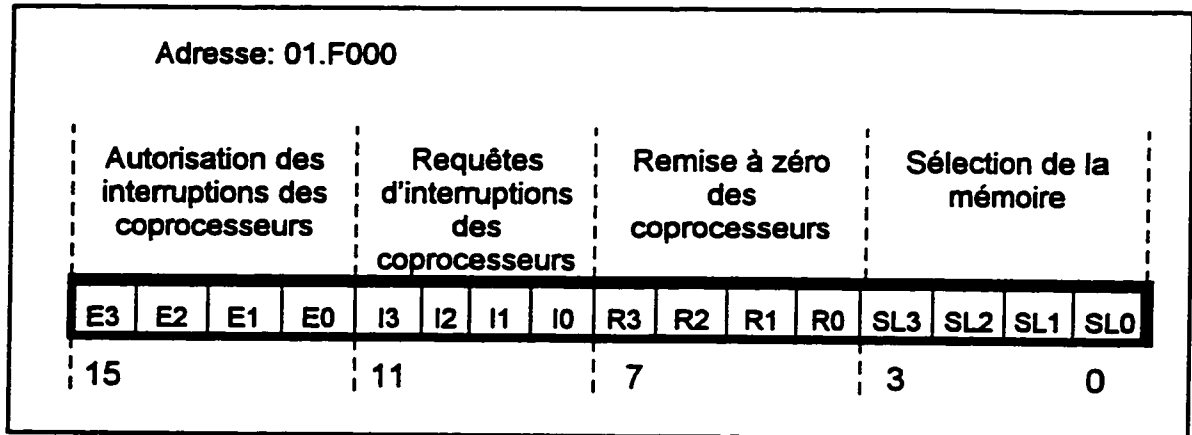


Figure 4.5 Registre d'état et de contrôle.

L'accès au bus maître - esclave se fait en utilisant deux registres adressables par micro-contrôleurs maître et esclaves. Le premier registre, nommé *registre principal*, est utilisé pour la réception et la transmission des données sur huit bits entre les microprocesseurs, alors que le deuxième registre, nommé *registre secondaire*, permet l'identification de la donnée. Ainsi, on identifie la tension ou le courant dans le cas de la réception des données par le μP maître ou on détermine si l'octet reçu au niveau des μP esclaves représente l'octet inférieur ou supérieur de la consigne (figure 4.6). Il faut souligner que le contenu du registre secondaire lu par le micro-contrôleur maître représente le complémentaire du registre secondaire visible par le micro-contrôleur esclave.

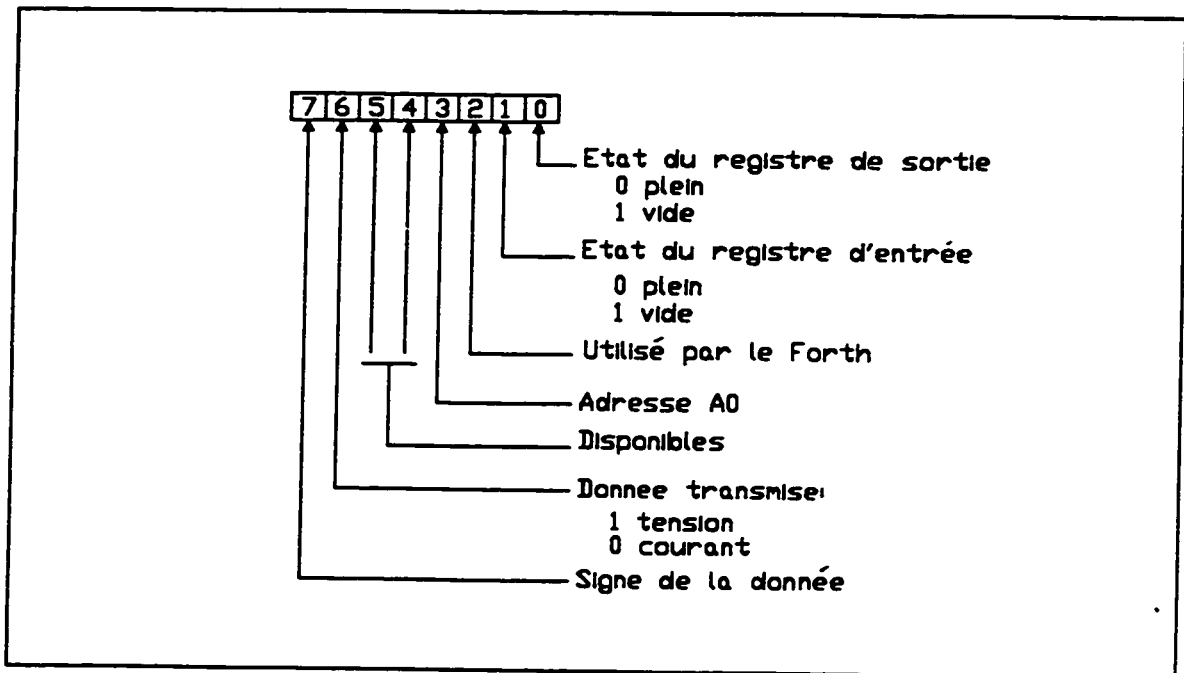


Figure 4.6 Registre secondaire du bus maître - esclave.

L'interruption INT0 détermine l'identité de l'esclave requérant, identifie la donnée disponible (tension ou courant en fonction du registre secondaire de communication), établit son signe et l'enregistre (sous-programme *Resc*). Après avoir servi un micro-contrôleur esclave, INT0 vérifie si celui-ci est déclaré actif. Les micro-contrôleurs esclaves sont déclarés actifs en mettant à un le bit correspondant de la variable « *pontact* ». Par exemple, le programme principal qui commande les ponts redresseurs 0 et 2 va assigner à la variable « *pontact* » la valeur 5. Cette valeur se représente en binaire sous la forme 0101 où les bits mis à un correspondent aux μP déclarés actifs. Si le μP requérant est actif, le programme de réglage est lancé après la fin de INT0. La

déclaration d'un pont comme étant inactif (par la mise à zéro de son bit de la variable « pontact ») est une façon d'ouvrir la boucle de réglage de tension d'un pont.

4.1.2 Sous-programmes de mesure et de protection

Les sous-programmes de mesure et protection font la mise à jour des données à la suite des interruptions produites par le matériel et assurent la protection du redresseur le cas échéant.

4.1.2.1 Protection rapide du courant

Le circuit de protection du courant (figure 3.4) provoque une demande d'interruption (INT2) chaque fois que le courant mesuré par le maître dépasse une valeur établie. Cette interruption permet une protection rapide du redresseur en cas de court-circuit ou de surcharge. Ce type de protection s'applique seulement pour la mesure de courant du maître.

La figure 4.7 montre les formes d'ondes enregistrées au moment où le courant de charge dépasse la limite maximale établie par le potentiomètre R9. On identifie, sur cette figure, le courant de charge (B) qui excède la limite maximale de courte durée du courant (A) déterminé par le potentiomètre R9. À la suite du

dépassement, le matériel génère une demande d'interruption sur le front tombant du signal (C).

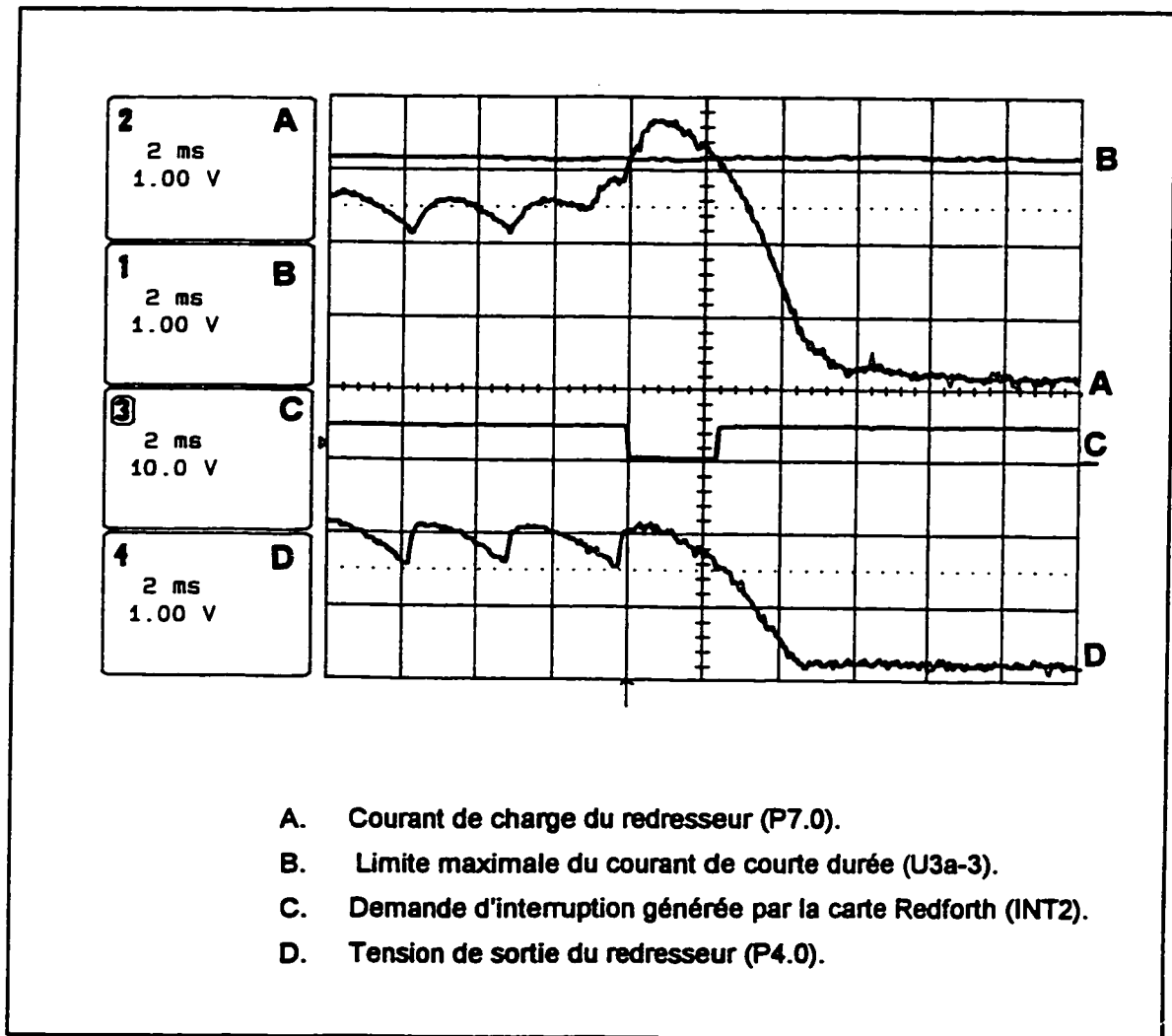


Figure 4.7 Protection de courant du micro-contrôleur maître.

L'interruption INT2 mesure le courant du maître, annule la transmission des commandes aux thyristors par la mise à zéro du port P5.3 et allume la diode

electroluminescente D12. Ensuite le message « Courant maximal du maître dépassé » apparaît à l'écran de l'ordinateur.

Pour bloquer un thyristor en conduction, sans l'aide d'un circuit d'extinction, il faut que le courant du thyristor devienne nul. Dans le cas d'un court-circuit à la sortie du redresseur, l'action de INT2 arrive trop tard pour les thyristors en conduction, mais elle peut sauver les thyristors qui devraient entrer en conduction. En effet, la figure 4.7 nous permet de remarquer que même si la commande est arrêtée, le courant de charge (B) continue à circuler jusqu'au passage par zéro de la tension de sortie (D). Il est important de protéger les thyristors qui ne sont pas encore en conduction étant donné leur grand nombre qui peut atteindre vingt-quatre.

4.1.2.2 Variation des paramètres extérieurs

Le circuit d'interface du maître génère une demande d'interruption INT1 si la variation du courant ou de la tension du réseau dépasse un taux déterminé par les potentiomètres R1, R2, R3 et R4 (figure 3.2). Le but de l'interruption INT1 est de lire les nouvelles données (tension du réseau, courant du maître) et de demander la mise à jour des réglages de tous les esclaves ou, s'il s'avère nécessaire, de déclencher la protection de tension.

La disparition d'une phase du système triphasé d'alimentation peut avoir des conséquences imprévisibles sur la tension de sortie du redresseur si elle n'est

pas détectée à temps. La figure 4.8 présente une première approche où le programme principal détecte la diminution de la tension moyenne du pont auxiliaire suite à la disparition d'une phase de la tension redressée (a) et arrête la commande du redresseur. On remarque que la tension redressée du pont (b) devient incontrôlable entre la disparition de la phase et l'arrêt du redresseur commandé par le port P5.3 (c), ce qui produit des impulsions de la tension de sortie plus grandes que la tension réglée.

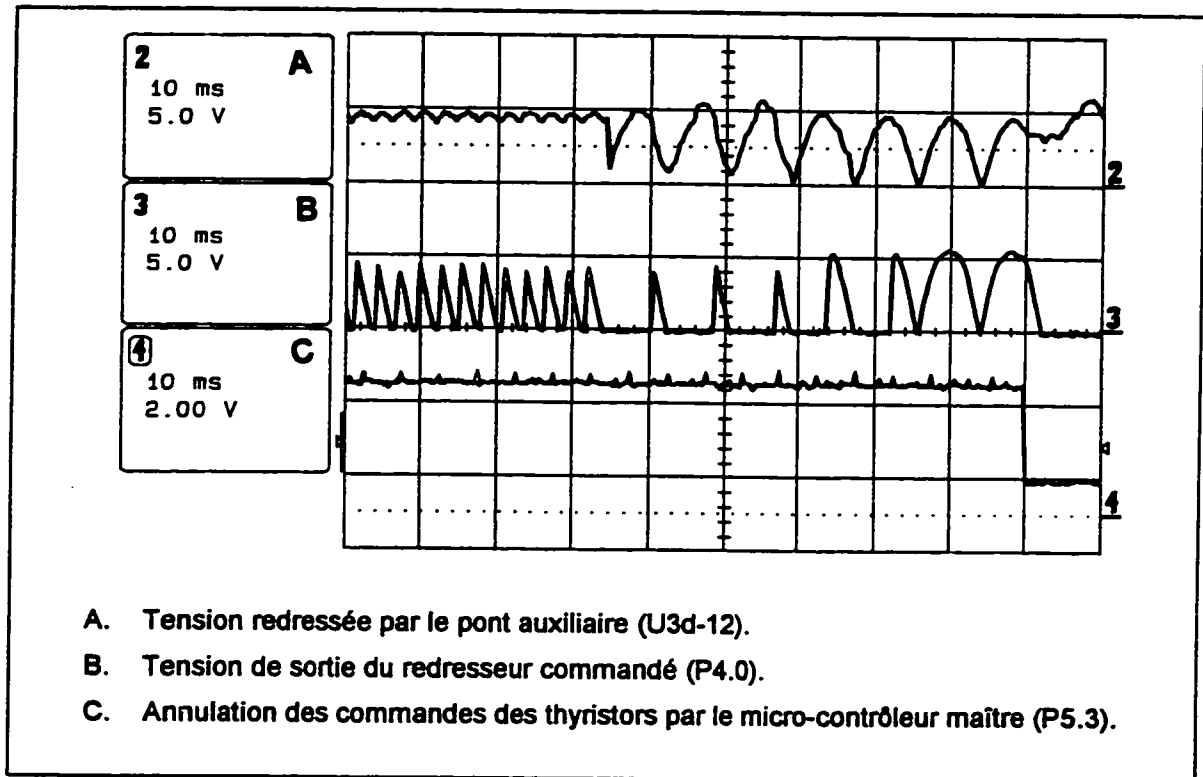


Figure 4.8 Disparition d'une phase d'alimentation: protection réalisée par le programme principal.

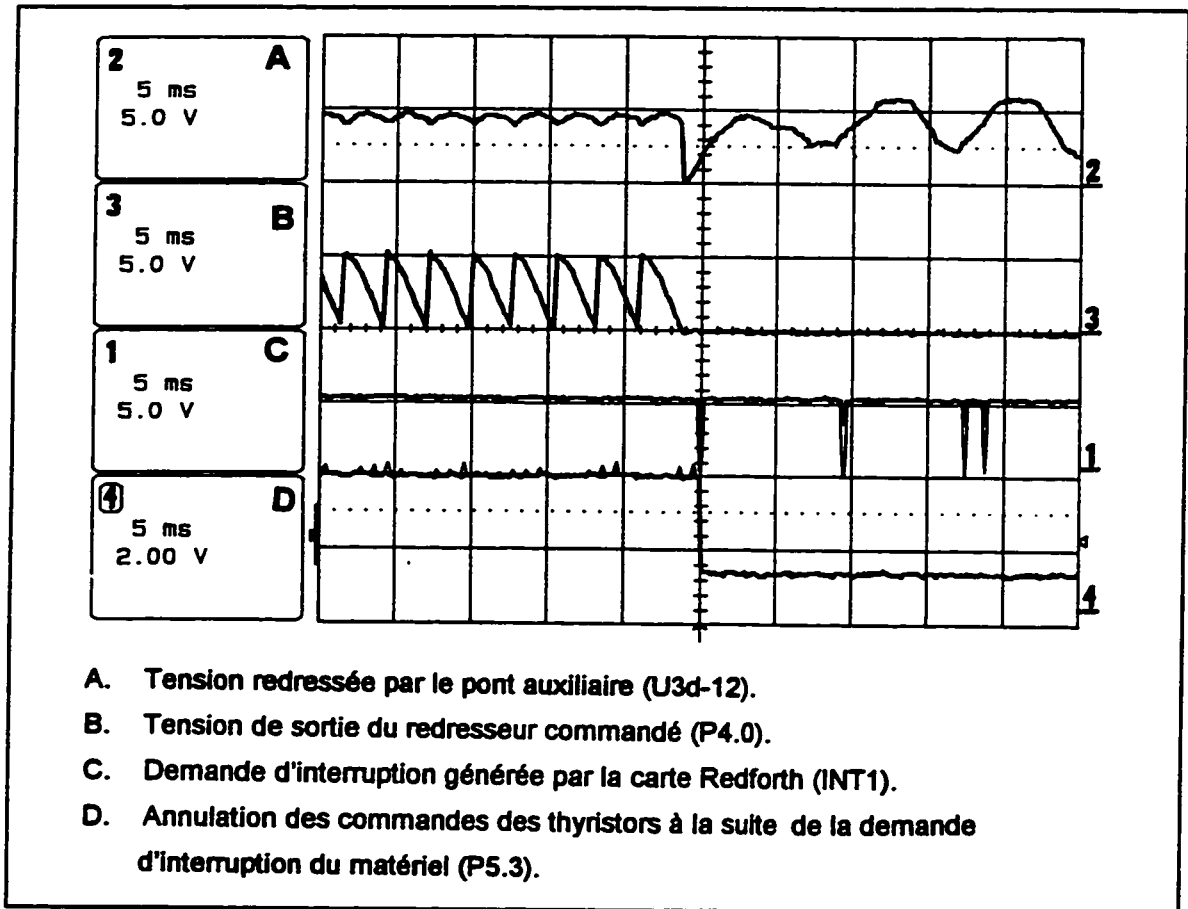


Figure 4.9 Disparition d'une phase d'alimentation: protection basée sur la détection de la variation de la tension du réseau par le matériel.

Une deuxième approche, qui a été retenue pour la version finale du programme, consiste à détecter la disparition d'une phase à l'aide du circuit de détection de la variation de la tension du réseau. Les possibilités de détection des variations de tension de ce circuit sont limitées par le reflet dans la tension du primaire de la commutation des thyristors du secondaire. La commutation des thyristors implique l'apparition d'une pointe de courant qui tend à annuler la tension du secondaire du transformateur. Les chutes de tension ont une forte

amplitude et une courte durée et elles se transmettent au primaire du transformateur. L'ajout d'un filtre passe-bas au montage présenté à la section 3.2.3 nous permet d'éliminer la détection des chutes de tension dues à la commutation des thyristors et de détecter seulement les variations de la tension du réseau qui ont une durée plus longue. La disparition d'une phase du système d'alimentation triphasée, montrée à la figure 4.9, est un exemple de fonctionnement du circuit de détection des variations de la tension de réseau. La tension redressée par le pont auxiliaire (a) provoque une interruption (c) à la suite de la disparition d'une phase d'alimentation. Le programme d'interruption va lire la valeur de la tension du réseau et, si la tension mesurée est trop basse, le micro-contrôleur maître va annuler la commande des thyristors à l'aide du port P5.3 (d). En conséquence, la tension redressée (b) deviendra nulle sans générer d'impulsions non contrôlées.

4.1.2.3 Protection de surcharge du courant mesuré par le micro-contrôleur maître

Le programme principal et le sous-programme de l'interruption INT1 vérifient l'amplitude du courant mesuré par le μP maître. La détection d'un courant qui dépasse la valeur du courant nominal lance le calcul du temps maximal de fonctionnement à ce régime et met en route la minuterie TA4. La fin du décompte de cette minuterie produit une demande d'interruption qui va exécuter le sous-programme « timer_courant ». La procédure de surveillance

du courant de surcharge une fois amorcée, le sous-programme d'interruption vérifie à chaque fois si le courant est revenu à la valeur normale. Si le courant demeure anormal jusqu'à l'écoulement du délai calculé par le programme principal, le système est arrêté, la diode electroluminescente correspondante (D11) est allumée continuellement et le message « Surcharge du maître » est transmis à l'ordinateur. Par contre, si le courant redevient normal, la minuterie s'arrête et le message « Surcharge du maître » affiché à l'écran de l'ordinateur disparaît.

4.1.2.4 Protection de courant des micro-contrôleurs esclaves

Le courant de charge de chaque redresseur commandé par les micro-contrôleurs esclaves est surveillé au niveau du maître par le sous-programme « *ddelais* ». Une fois que le micro-contrôleur maître détecte un courant de surcharge, il va mettre en route la minuterie TB2. Le programme d'interruption associé au passage par zéro du compteur de la minuterie « *timer_courant_esc* » vérifie la valeur du courant. La structure du programme est similaire au programme de protection du courant mesuré par le maître (« *timer_courant* »). Le μ P maître arrête le système, allume la diode D12 et transmet à l'ordinateur le message « Surcharge de longue durée des esclaves » si le temps prescrit par « *ddelais* » s'est écoulé et si le courant n'est pas redevenu normal.

4.1.3 Sous-programmes de signalisation

L'état du système est signalé par les diodes electroluminescentes D9 à D14. Les situations anormales entraînent le clignotement des diodes electroluminescentes (identification au paragraphe 3.2.4) produit par le programme d'interruption « tclin » de la minuterie TA3. La minuterie s'arrête si le système revient aux conditions normales de fonctionnement.

Les situations qui génèrent l'arrêt du système forcent la diode correspondante à rester allumée continuellement.

Le sous-programme « clin » reçoit sur la pile de données le numéro de la diode (2 à 7) qui doit clignoter. Il met en route la minuterie si elle ne l'est pas encore et fait clignoter la diode.

Le sous-programme « StopClin » éteint et arrête le clignotement la diode désignée par le numéro (2 à 7) reçu sur la pile de données.

4.1.4 Sous-programmes de commande

La boucle de réglage de tension est fermée par les sous-programmes de commande qui calculent les consignes d'angle pour des micro-contrôleurs esclaves en fonction des données transmises par ceux-ci et des mesures faites par le micro-contrôleur maître.

4.1.4.1 Sous-programme « norm »

Le sous-programme « norm » protège le système contre une tension de réseau en dehors des limites. Une tension anormale détermine l'arrêt du système, la signalisation lumineuse (D14) et l'affichage à l'écran de l'ordinateur personnel du message correspondant. Le sous-programme compare les nouvelles données mesurées ou reçues avec leurs anciennes valeurs et si les variations dépassent le facteur de sensibilité, « k_s », alors les nouvelles données seront transmises à l'ordinateur.

4.1.4.2 Sous-programme *principal*

Le programme principal « main » est exécuté à la suite de la requête d'interruption de la minuterie TB2. Cette minuterie n'est pas mise en route, mais nous utilisons son drapeau indiquant que l'interruption est requise avec son vecteur d'interruption qui lui est associé. C'est pourquoi on peut lancer l'exécution du programme principal à partir des programmes ayant une haute priorité (enregistrés à 0INT, 1INT, 2INT) sans pour autant affecter son niveau de priorité qui est plus bas. Effectivement, les sous-programmes de réception de données et ceux de protection du système mettent active la demande d'interruption du programme principal et le programme de réglage est lancé à la fin de l'interruption courante.

Au début, le programme principal identifie le pont pour lequel il va faire le réglage à l'itération courante. Ainsi, selon la valeur de la variable « pont », le programme va transmettre la consigne au pont concerné en utilisant pour le calcul les données correspondantes. Ensuite, le programme vérifie s'il y a eu des variations de la consigne ou des données mesurées par les micro-contrôleurs maître et esclaves. Les données qui ont changé sont enregistrées dans la matrice « état » et le programme de transmission, moins prioritaire, va se charger de leur transmission à l'ordinateur personnel. La tâche de base du programme principal consiste à fermer la boucle de réglage de tension en calculant la consigne des micro-contrôleurs esclaves à partir de la matrice « comm » qui contient les résultats de la fonction:

$$N = K_{\alpha} \arccos \frac{U_{\text{sortie}}}{U_{\text{réseau}}} \quad (5.1)$$

$U_{\text{réseau}}$ est la tension du réseau. Elle varie de $U_{\text{réseau}} \text{ minimale}$ à $U_{\text{réseau}} \text{ maximale}$. L'écart entre ces valeurs est divisé en 1F (hexadécimal) intervalles qui donnent la dimension verticale de la matrice « comm ». K_{α} correspond au coefficient de proportionnalité entre les angles électriques et la consigne des esclaves. U_{sortie} représente la consigne de la tension de sortie qui varie de 0 à FF hex. Sa variation donne la dimension horizontale de la matrice « comm ».

La différence entre la valeur de la tension de sortie et la consigne représente l'erreur de réglage et elle est passée à travers un régulateur (figure 4.10). Le résultat est ajouté au point de fonctionnement qui a été calculé pendant l'itération précédente et qui caractérise l'état actuel du redresseur. Le point de fonctionnement permet le calcul de l'indice horizontal de la matrice « comm ». La tension du réseau mesurée et la tension du réseau minimale admissible servent au calcul de l'indice vertical de la matrice « comm ». Les deux indices pointent sur la valeur de la consigne des esclaves. L'indice horizontal représente une approximation du point de fonctionnement. Pour une meilleure précision, on calcule une correction fine qui représente la différence entre le point de fonctionnement calculé (la variable « index_h ») et l'approximation du point de fonctionnement (la variable « index ») exprimée en unités de consigne. La consigne envoyée aux esclaves est l'addition de la valeur obtenue précédemment de la matrice « comm », de la correction du déséquilibre des ponts (utilisée seulement pour la configuration des ponts en parallèle), de la correction dépendante du régime du convertisseur (redresseur ou onduleur) et de la correction fine. Cette approche du programme principal est destinée à la commande d'un redresseur en utilisant une seule boucle de réglage. L'avantage de ce programme est de permettre la diminution des dimensions de la matrice « comm ». Le désavantage de cette approche est un temps de

long service du programme principal qui détériore fortement la dynamique d'un système basé sur quatre boucles de réglage de tension indépendantes. La solution pour ce problème consiste à réduire le temps de service du programme principal. En effet, nous avons conçu une version du programme principal qui laisse au sous-programme de transmission des données à l'ordinateur la tâche d'identifier les variations des données qui doivent être transmises et nous avons simplifié le calcul du pont de fonctionnement du redresseur. Ainsi, on a éliminé l'approximation linéaire de la consigne et la résolution du réglage dépend des dimensions de la matrice « comm ». La réponse à une entrée échelon des systèmes utilisant ce programme sera présentée à la section 4.3.

4.2 Sous-programmes des micro-contrôleurs esclaves

Les tâches des micro-contrôleurs M37450 sont: la commande d'un pont de Graetz (ou l'équivalent de deux étoiles triphasées), la réception de la consigne d'angle et la transmission des nouvelles données (tension et courant de sortie) au maître. La figure 4.11 présente l'architecture du système esclave.

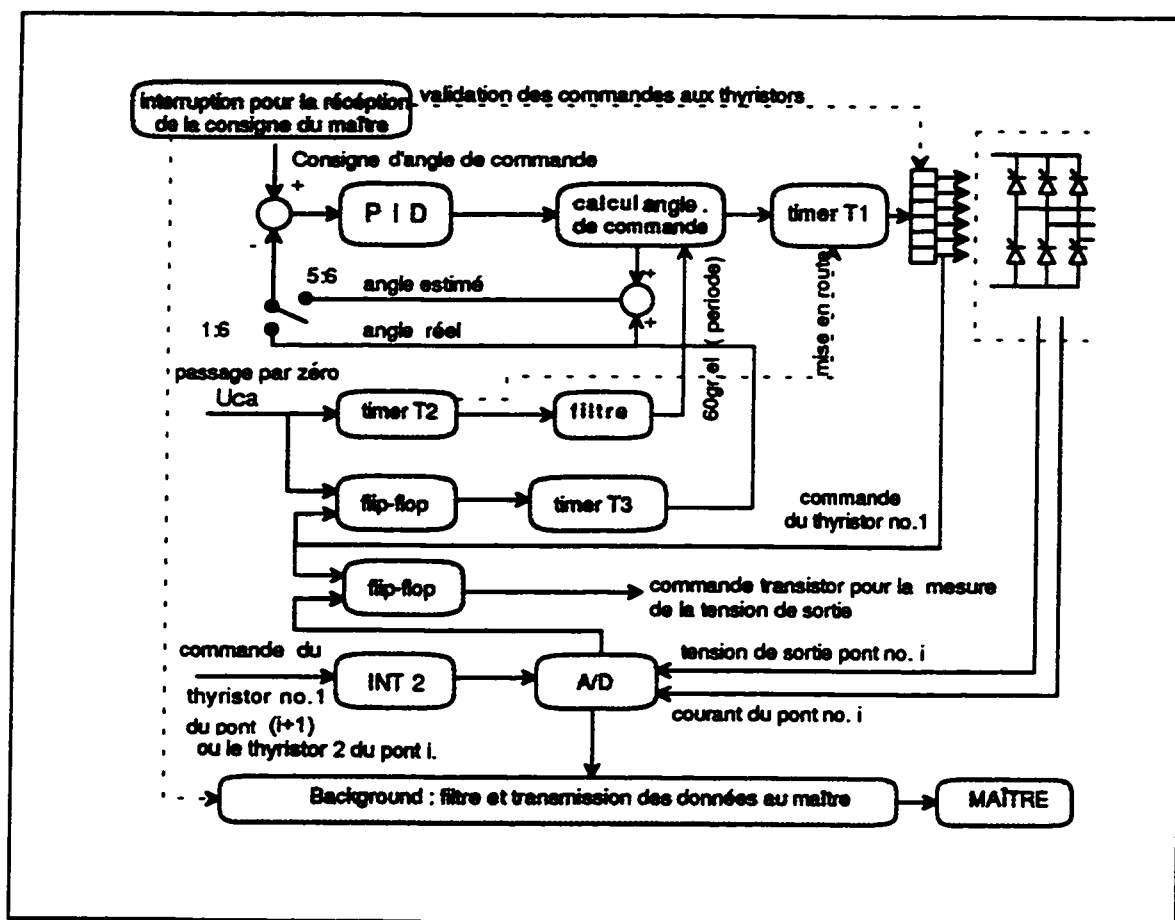


Figure 4.11 Architecture du logiciel du micro-contrôleur esclave.

La commande des thyristors est synchronisée avec la tension du réseau au début de chaque période afin de prévenir la perte de la synchronisation, provoquée par l'accumulation d'erreurs de mesure d'angles de commande de thyristors. L'utilisation de la période mesurée du réseau dans les calculs permet d'éliminer les erreurs dues aux petites variations de la fréquence du réseau.

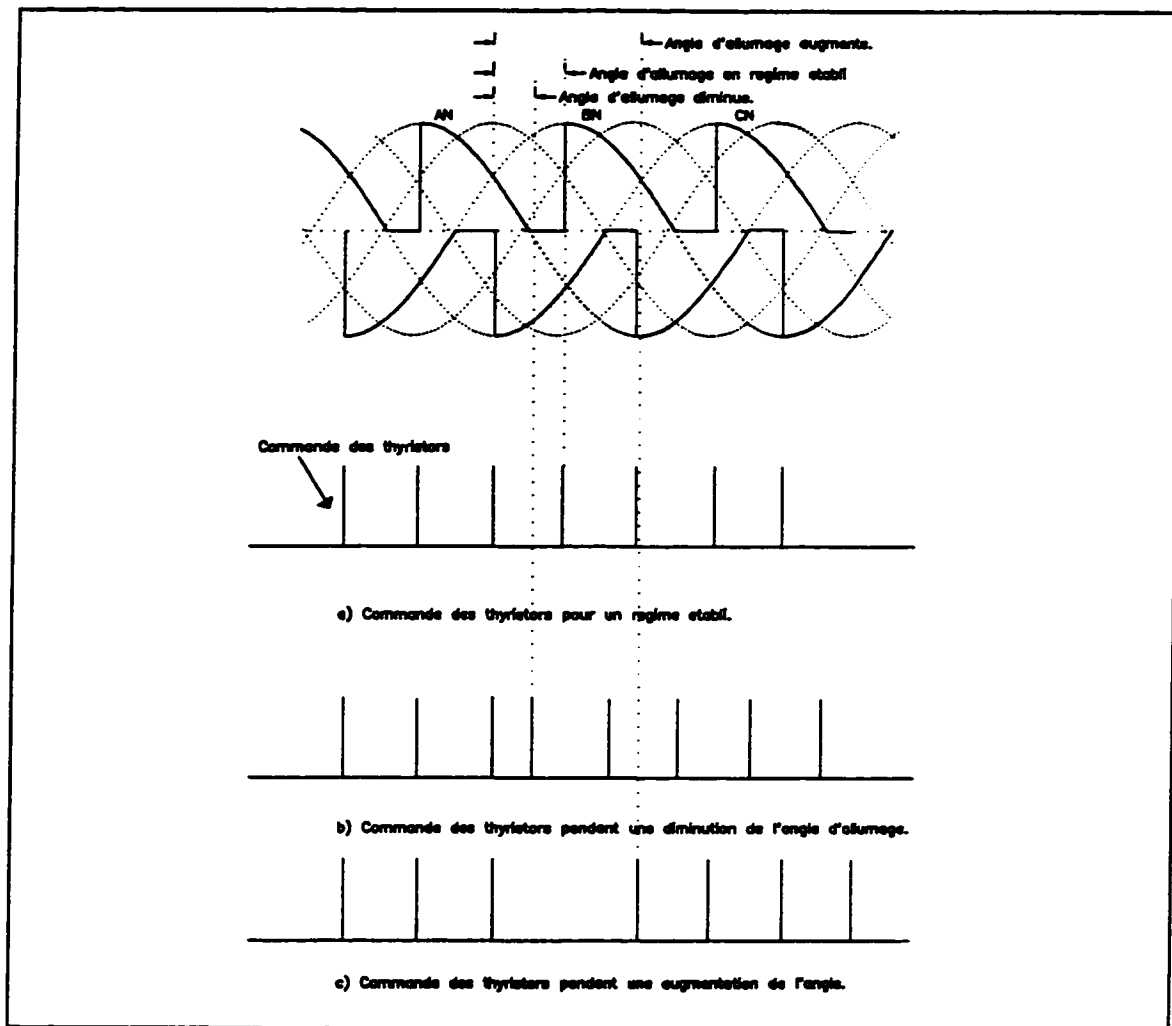


Figure 4.12 Commande équidistante des thyristors.

Pour la commande des thyristors, on peut utiliser soit le calcul individuel des impulsions de commande, souhaitable si on doit corriger l'asymétrie du réseau triphasé, soit la méthode des impulsions équidistantes [4] qui est plus simple et permet d'avoir une bonne dynamique du système. La méthode de commande retenue est celle à impulsions équidistantes. Dans cette méthode, le calcul du

moment d'allumage des thyristors se fait toujours par rapport à l'ancienne commande transmise aux thyristors.

Pour un régime établi, les impulsions de commande des thyristors sont équidistantes et elles se succèdent à tous les 60° (figure 4.12a). Pour diminuer l'angle de commande, on peut réduire le délai entre les impulsions des thyristors jusqu'à un minimum de 15° (figure 4.12b). Si on ne réussit pas à obtenir la consigne désirée après une seule itération, on répète la diminution de l'angle pour les intervalles suivants. Si on désire agrandir l'angle de commande, il suffit d'augmenter le délai entre deux impulsions de thyristors (figure 4.12 c).

4.2.1 Sous-programmes de communication

Les sous-programmes de communication permettent la réception de la consigne du micro-contrôleur maître ainsi que la transmission à celui-ci des données mesurées par le micro-contrôleur esclave.

4.2.1.1 La réception de la consigne du micro-contrôleur maître

Le micro-contrôleur maître envoie sa consigne au micro-contrôleur esclave sous forme de deux octets consécutifs: les octets inférieur et supérieur. Le statut de l'adresse A0 du micro-contrôleur maître permet d'identifier l'octet reçu.

Ainsi, A0=1 correspond à la réception d'un octet inférieur de la consigne, tandis que A0=0 indique la réception d'un octet supérieur de celle-ci. L'état de A0 est lu par le micro-contrôleur esclave dans le registre secondaire et l'octet reçu est lu dans le registre principal du bus des micro-contrôleurs maître - esclaves.

L'octet supérieur contient soit le deuxième octet de la consigne soit les mots de commande envoyés par le micro-contrôleur maître. Les mots de commande sont des octets qui ont une valeur plus grande que 76 (hexadécimal). L'arrivée d'un octet dans le registre de réception du micro-contrôleur esclave génère une demande d'interruption qui aura le plus haut niveau de priorité associé aux requêtes d'interruption du μP esclave. Il faut rappeler que le niveau de priorité des interruptions est fixe et est déterminé par la structure interne du μP . À la suite de la requête d'interruption, le sous-programme « InBuf » lira l'octet et enregistrera la consigne ou exécutera la commande selon le cas. Les mots de commande reconnus par le micro-contrôleur esclave sont les suivants:

- la combinaison FF arrête les commandes des thyristors pour chaque pont individuellement. Cet arrêt des thyristors prend plus de temps à être exécuté que l'arrêt général des thyristors commandé par le micro-contrôleur maître au port P5.3.

- l'octet FE démarre la commande des thyristors qui est arrêtée à la suite de la commande FF. La commande est inefficace si l'arrêt général des thyristors est activé (diode D11 allumée).
- le mot de commande FD active la transmission périodique des données mesurées au micro-contrôleur maître.
- la commande FC annule la transmission périodique des données mesurées au micro-contrôleur maître.
- le mot de commande FB provoque la transmission immédiate de données au micro-contrôleur maître. Le μ P maître peut ainsi forcer un micro-contrôleur esclave à rapporter les données mesurées même s'il n'y a pas eu de variations.

4.2.1.2 Transmission des données au micro-contrôleur maître

La mesure et la transmission des données sont faites une fois par période par les sous-programmes « ADInt » et « transmit ». Le programme « transmit » qui s'exécute dans l'arrière plan au plus bas niveau de priorité, filtre numériquement les nouvelles données et vérifie ensuite si le résultat diffère des dernières données transmises au micro-contrôleur maître. Cependant, il est nécessaire d'avoir une réponse rapide du système lors d'une variation importante de la mesure afin d'assurer une bonne dynamique de la réponse du redresseur. La solution consiste à laisser passer sans filtrage les grandes

variations. Le filtre utilisé pour des petites variations évite un flux inutile d'informations entre les micro-contrôleurs esclave et maître qui provoquera l'épuisement du temps de travail du micro-contrôleur maître. Un filtre ayant une constante de temps trop grande peut créer un manque d'information pour le maître et donc l'inactivité de la boucle de réglage. Afin d'éviter le blocage du maître causé par un manque de données, le maître peut ordonner aux microprocesseurs esclaves la transmission périodique des données.

4.2.2 Sous-programmes de mesure

Les sous-programmes de mesure assurent au système les données nécessaires à son fonctionnement. Ainsi, le micro-contrôleur esclave mesure l'angle de commande des thyristors, la période de la tension du réseau, la tension et le courant de charge du redresseur commandé.

4.2.2.1 Mesure de l'angle de commande

La mesure de l'angle de commande des thyristors, nécessaire à la boucle de réglage d'angle, se fait à l'aide de la minuterie T2 du μP esclave programmée en mode "mesure d'impulsion". L'angle de commande mesuré représente le délai entre le passage par zéro de la tension de référence U_{CA} et la commande du thyristor 1. La mesure se fait une fois par période pour l'angle de commande du thyristor 1 tel que vu dans la section 3.3.3. La fin de la mesure produit une

requête d'interruption qui démarrera l'exécution du sous-programme « EV3 ». Le but de « EV3 » est d'enregistrer le contenu du registre de la minuterie T2 qui représente l'angle mesuré.

4.2.2.2 Sous-programme de mesure de la période de la tension du réseau

L'interruption de la minuterie T3, programmée en mode « mesure d'impulsion », mesure une fois par période le délai entre deux passages par zéro de la tension U_{ca} . Le sous-programme d'interruption, « EV2 », qui apparaît à la fin de chaque mesure, enregistre et filtre numériquement la nouvelle valeur de la période.

4.2.2.3 Début de la mesure de tension

La mesure échantillonnée de la tension de sortie est nécessaire lorsqu'on désire déterminer la contribution relative à la tension de sortie d'un pont par rapport à la tension des autres ponts dans un redresseur complexe (tel qu'un redresseur à vingt-quatre phases). Ceci permet de compenser les différences en corrigeant l'angle de commande. Cette méthode de mesure implique la nécessité de synchroniser le début de l'intégration de la tension de sortie (présentée à la section 3.3.1) et la mesure de cette tension à la période de conduction des thyristors. Ainsi, le début de l'intégration est toujours déterminé par le logiciel et il coïncide avec la mise en conduction du thyristor 1 du pont surveillé. Le moment où on mesure la tension est déterminé par le premier thyristor suivant qui entre en conduction. La synchronisation avec ce thyristor

se fait à l'aide de l'interruption « INT2 » qui commande la mesure de la tension de sortie. L'interruption peut être générée par la commande d'un thyristor ou par tout autre événement externe. Par exemple, si on désire passer à la commande des ponts de Graetz indépendants, il suffit d'utiliser comme source d'interruptions le thyristor 2 du même pont. Il est nécessaire de étalonner le circuit de mesure de la tension de sortie à chaque fois qu'on change l'intervalle de mesure parce que l'intervalle d'intégration change aussi. Une autre solution, qui s'applique seulement à la commande des ponts de Graetz indépendants, consiste à commander la mesure en utilisant le logiciel des micro-contrôleurs esclaves.

Une autre methode serait de mesurer la tension de sortie sans échantillonnage. Dans ce cas, le circuit d'échantillonnage reste toujours inactif et le μP mesure directement la tension continue de sortie.

4.2.2.4 Sous-programme de mesure de la tension

Tel que vu au paragraphe précédant, la mesure de tension de sortie peut être accomplie par lecture échantillonnée ou continue. La mesure de la tension et du courant est effectuée par le sous-programme « ADINT » qui utilise le convertisseur A/N à 8 bits du microprocesseur M37450. Selon la première approche présentée au paragraphe 4.2.2.3, la mesure est déclenchée de

l'extérieur par l'interruption INT2. Ensuite, la fin de la mesure de tension met en route la mesure du courant.

4.2.3 Sous-programme de commande

Les thyristors sont commandés par le programme « TA4INT » exécuté par l'interruption que génère la minuterie TA4 programmée en mode « minuterie de temps ». Le sous-programme, exécuté six fois par période, détermine le délai entre deux commandes consécutives d'amorçage des thyristors. Ainsi, le sous-programme calcule la différence entre la consigne du micro-contrôleur maître et l'angle mesuré, et la passe à travers un régulateur P (figure 4.10). Une fois sur six « TA4INT » détermine le délai jusqu'à la prochaine commande en utilisant un angle de commande qui est mesuré une fois par période, tel que vu au paragraphe 3.3.3. Les cinq itérations, suivant le calcul fait à la première, utilisent un angle déterminé à partir du dernier angle mesuré et des corrections de la commande qui ont été calculées. L'angle évalué permet ainsi de fermer la boucle de réglage cinq fois sur six. La sortie du régulateur et la mesure de la période de la tension servent à l'évaluation du délai entre les commandes des thyristors. Le délai obtenu à la suite de l'évaluation est chargé dans le registre de la minuterie T1. L'ordre de commande approprié pour les thyristors est déterminé par le sous-programme « cmdthy » qui utilise la table « comth ».

L'ordre de commande est transmis aux thyristors selon une procédure expliquée dans la section 3.3.3.

4.3 Réponse du système de réglage à une entrée échelon

Nous avons testé le fonctionnement du système de réglage pour trois configurations de la commande des redresseurs tel que la commande de quatre ponts de Graetz indépendants, la commande d'un pont dodécaphasé et la commande d'un redresseur à vingt-quatre phases.

Premièrement, on a réalisé la commande de quatre redresseurs indépendants (figure 2.7) où chaque micro-contrôleur esclave mesure la tension de sortie et le courant de charge de son pont. Le système de réglage de cette configuration se base sur une boucle de réglage d'angle au niveau de chaque micro-contrôleur esclave et quatre boucles de réglage de tension gérées indépendamment par le micro-contrôleur maître. Les boucles d'angle assurées par les micro-contrôleurs esclaves doivent être plus rapides que les boucles de tensions afin d'assurer la stabilité du système.

Le temps d'exécution du programme de réglage limite le délai minimal entre les commandes de thyristors. Par conséquent, nous avons choisi pour la boucle d'angle un régulateur simple de type P ($K_p = 0.25$) qui a un temps d'exécution court.

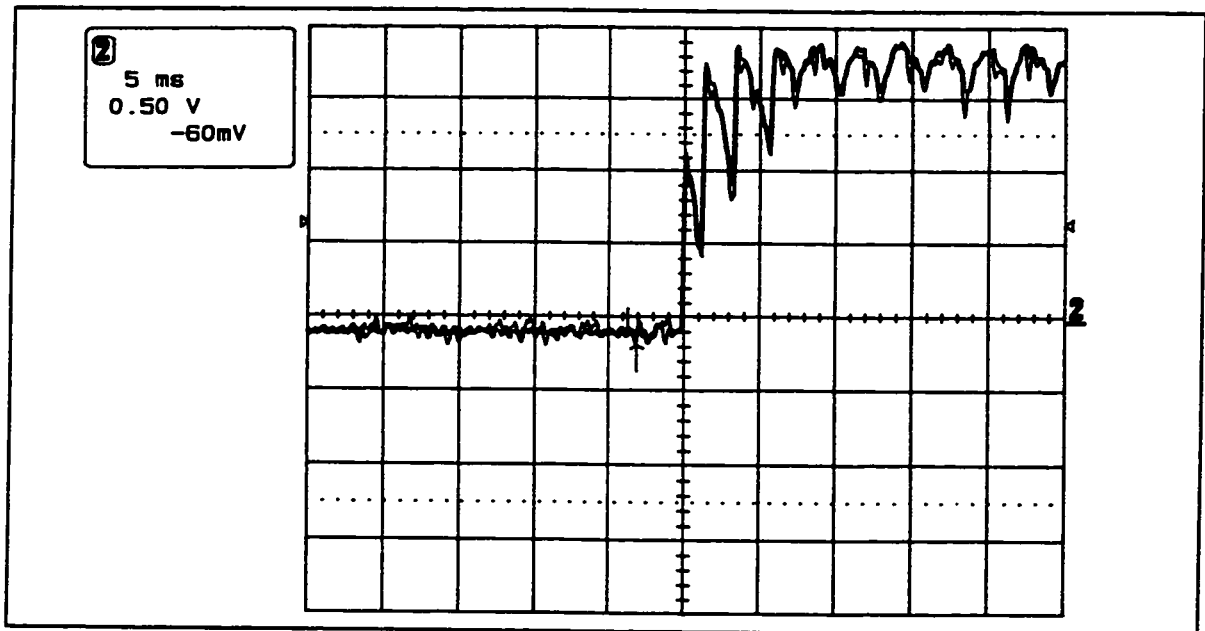
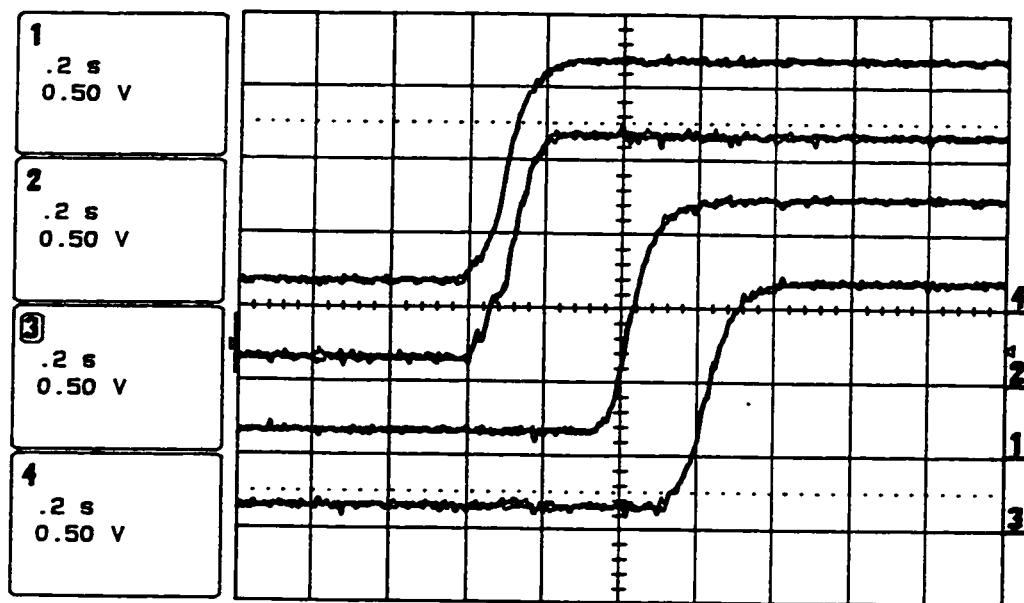


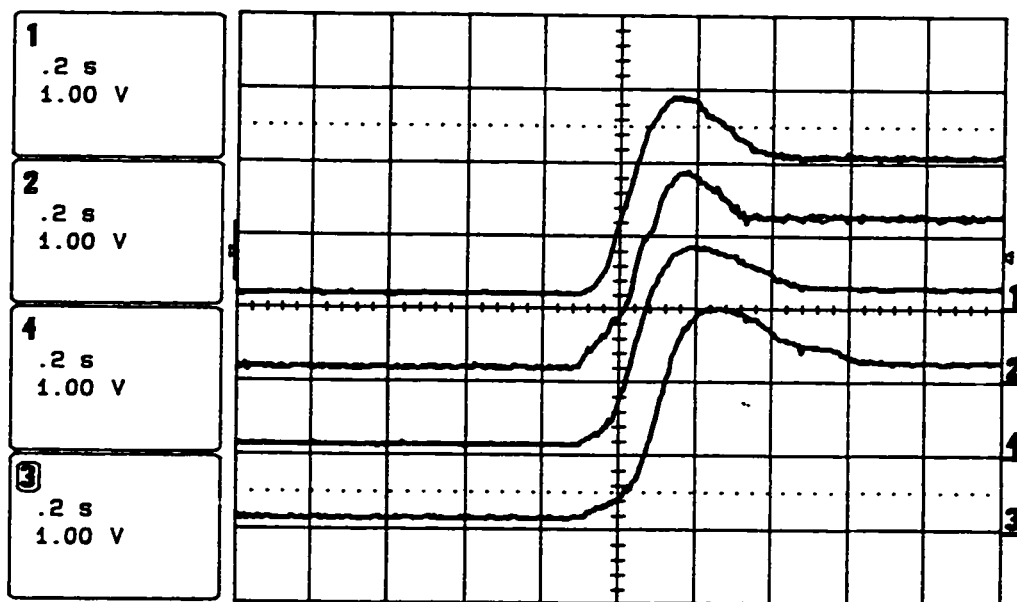
Figure 4.13 Réponse de la boucle d'angle à une consigne de type échelon.

La réponse de la boucle d'angle est indépendante de la configuration du redresseur commandé et nous l'avons obtenu en ouvrant la boucle de tension et en imposant un changement de consigne. Par exemple, pour évaluer le temps de réponse de la boucle de réglage d'angle du pont 1 on le déclare inactif (le bit 1 de la variable « pontact » est mis à zéro), on change la valeur de la consigne et ensuite on le déclare requérant (le bit 1 de la variable « req » est mis à un). La figure 4.13 présente la réponse à une consigne de type échelon de la boucle d'angle d'un micro-contrôleur esclave.

La réponse de la boucle de tension, utilisant un régulateur de type P, est présentée à la figure 4.14 a, b pour deux situations distinctes. Premièrement, nous avons changé simultanément la consigne de tous les redresseurs sans



A. Les micro-contrôleurs esclaves ne sont pas déclarés requérants ($K_p=0.125$).



B. Tous les micro-contrôleurs esclaves sont déclarés requérants ($K_p=0.25$).

Figure 4.14 Réponse des boucles de tension à une consigne de type échelon pour la commande de quatre ponts indépendants.

pour autant les déclarer comme requérants (les bits de la variable « req » sont mis à zéro). Nous observons à la figure 4.14.a que au fur et à mesure que les micro-contrôleurs esclaves transmettent leurs données au μP maître, les nouvelles consignes sont lues et le réglage pour chaque pont est mis en route. Par contre, la figure 4.14b présente la réponse des boucles d'angle à une consigne qui a déclaré tous les ponts requérants. Bien que, le réglage de la tension de sortie des redresseurs a commencé simultanément à la réception de la consigne, la dynamique des quatre réponses est différente selon le niveau de priorité assigné à chaque pont. Les deux approches présentées s'appliquent avec des bons résultats à la commande des redresseurs indépendants. Cependant, si on désire commander des configurations complexes qui impliquent la commande de plusieurs micro-contrôleurs esclaves, le réglage asynchrone des boucles de tension, selon la figure 4.14a, et la dynamique différente des réponses, selon la figure 4.14b, vont détériorer la réponse générale du convertisseur. La solution qui s'impose est l'utilisation d'une seule boucle de réglage de tension pour chaque redresseur commandé.

La deuxième configuration testée est celle d'un redresseur à douze phases constitué par deux ponts de Graetz en série. Le pont dodécaphasé représente une configuration de test intermédiaire entre la commande d'un pont de Graetz et la commande d'un pont à vingt-quatre phases. La figure 4.15 présente la réponse des boucles d'angle et leur effet sur la tension de sortie du redresseur.

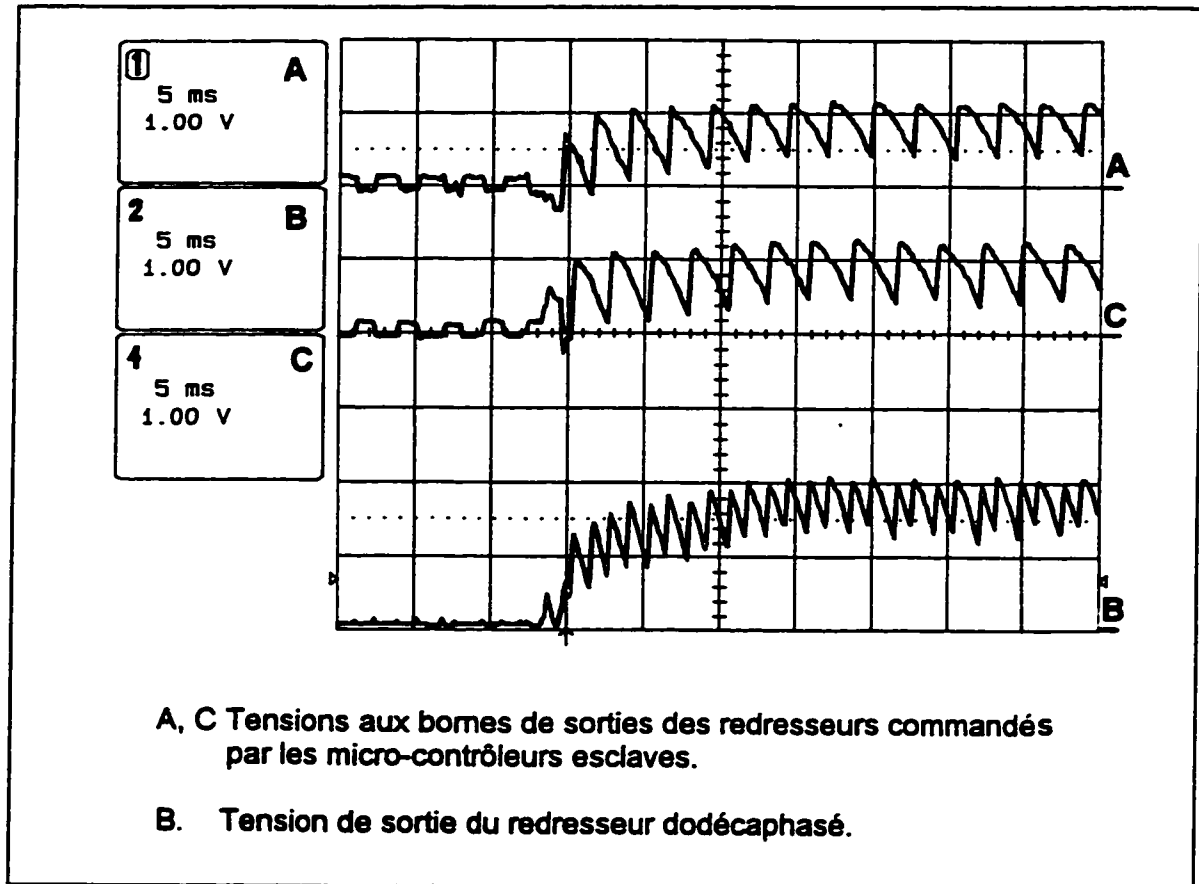


Figure 4.15 Réponse à une consigne de type échelon des boucles d'angle d'un redresseur dodécaphasé.

On observe qu'après six itérations la tension de sortie atteint 80% de la tension prescrite par la consigne ce qui correspond à la réponse déterminée par un régulateur P ayant $K_p=0,25$. Le redresseur dodécaphasé réalise le réglage en utilisant une seule boucle de tension qui calcule une consigne unique pour les deux ponts de Graetz. La figure 4.16 montre la réponse à une entrée échelon de la boucle de tension du redresseur dodécaphasé. Les courbes A et B représentent les tensions de sortie de chaque pont et la courbe C représente la

tension de sortie du redresseur dodécaphasé. Nous avons obtenu les courbes en déclarant actif le pont qui fait la mesure de la tension de sortie.

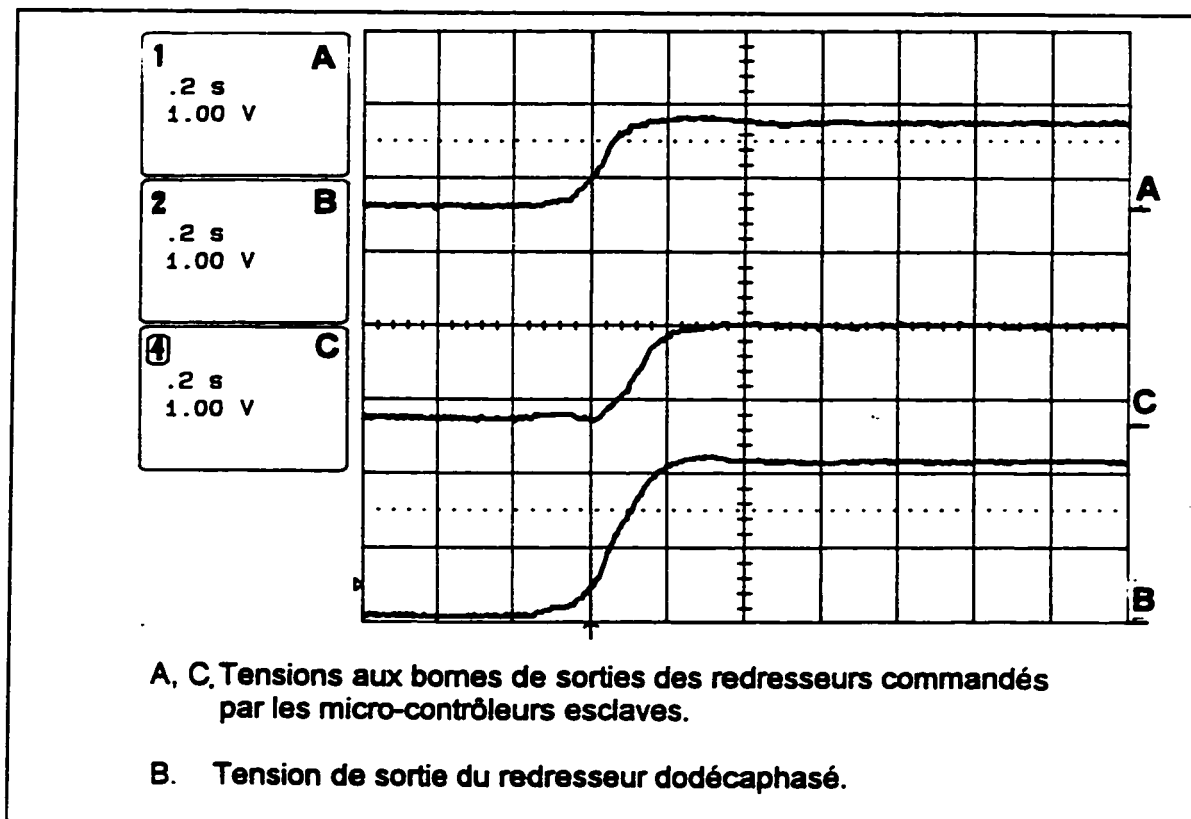


Figure 4.16 Réponse à une consigne de type échelon de la boucle de tension d'un redresseur dodécaphasé.

La dernière configuration testée est celle d'un redresseur série à vingt-quatre phases. La figure 4.17 montre la réponse des boucles d'angles à une entrée échelon. Les courbes A, B, C et D représentent trois des quatre tensions

mesurées aux bornes de sortie des ponts ainsi que la tension de sortie du redresseur à vingt-quatre phases.

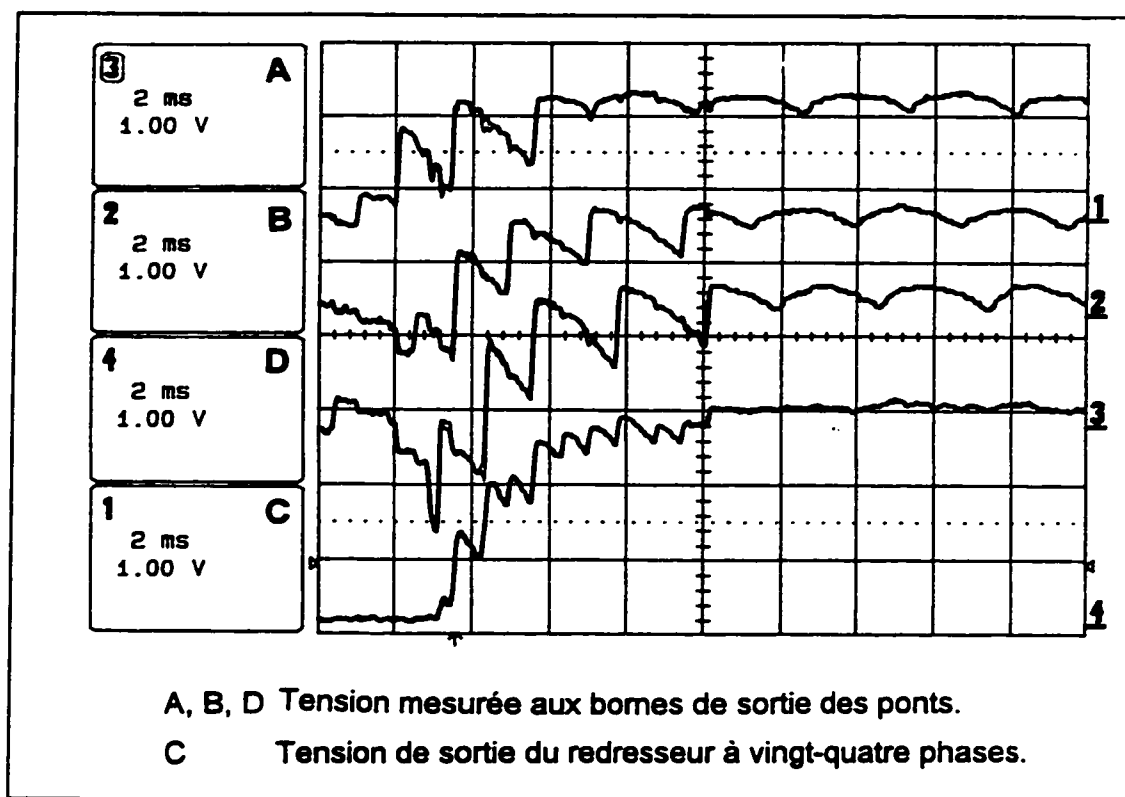


Figure 4.17 Réponses à une consigne de type échelon des boucles d'angle d'un redresseur à vingt-quatre phases.

La figure 4.18 montre la réponse des boucles de tension lorsqu'on applique un signal échelon à l'entrée.

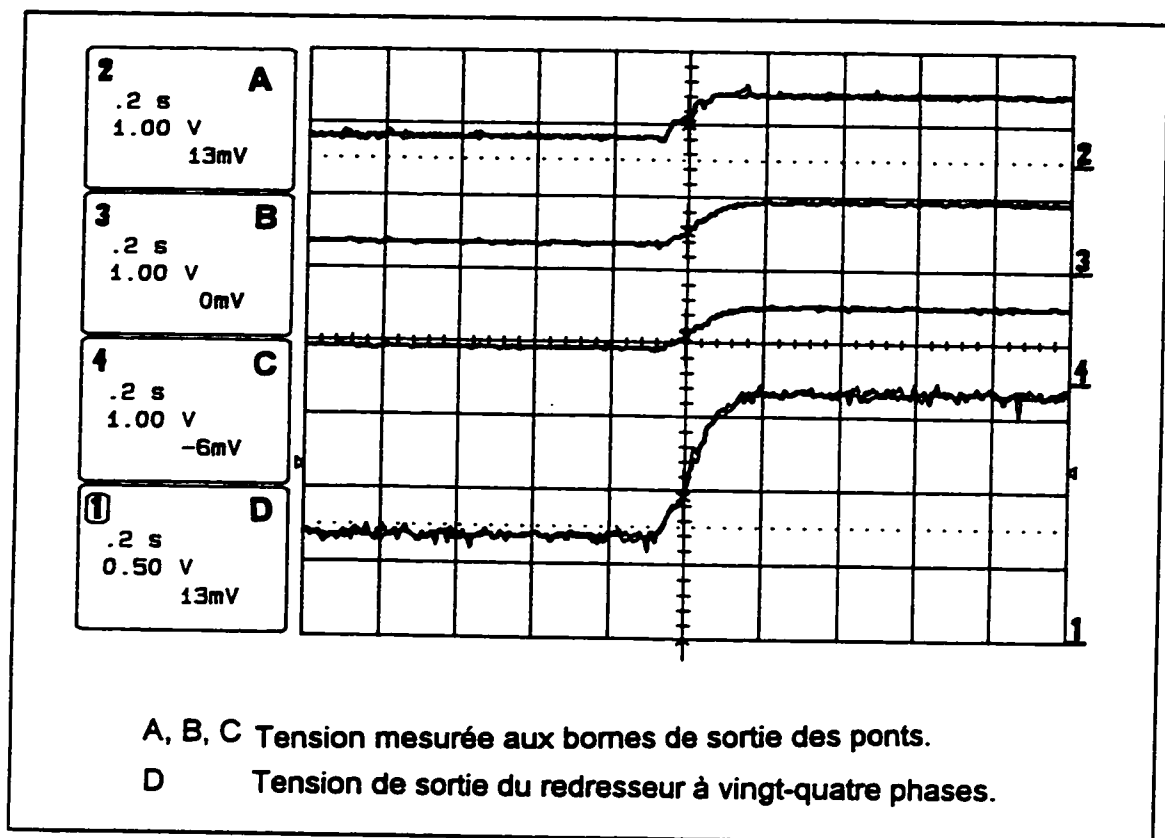


Figure 4.18 Réponses à une consigne de type échelon des boucles de tension d'un redresseur à vingt-quatre phases.

Dans le quatrième chapitre nous avons présenté la structure et les fonctions des programmes des micro-contrôleurs maître et esclaves. La majorité des sous-programmes s'exécutent à la suite des demandes d'interruptions générées par les minuteries des microprocesseurs ou par la carte d'interface Redforth. Nous avons testé plusieurs configurations qui nous ont confirmé l'influence du temps d'exécution des sous-routines et de leurs niveaux de priorité sur la dynamique du système. Les tests effectués démontrent les capacités de réglage du système de commande conçu.

5. CONCLUSION

Ce mémoire de maîtrise a présenté les principales étapes qui ont contribué à la rencontre des objectifs fixés. Le système réalisé permet la commande d'une série de configurations de redresseurs commandés qui serviront au développement de nouvelles configurations de redresseurs.

Pour atteindre nos objectifs, nous avons développé une carte d'interface capable de commander jusqu'à vingt-quatre thyristors et qui mesure cinq tensions et cinq courants, tout en assurant les protections de base d'un redresseur: tension en dehors de limites fixées, détection de l'absence d'une phase d'alimentation, protection rapide de courant et protection du courant de surcharge.

Ensuite, nous avons développé un logiciel de commande pour la carte micro-ordinateur écrit en Forth et Assembleur qui réalise le réglage du redresseur en utilisant jusqu'à quatre boucles de réglage d'angle et quatre boucles de réglage de tension. La structure du logiciel permet de passer facilement d'une configuration de commande à une autre.

Le potentiel de la carte d'interface et du logiciel de commande est confirmé par les tests faits sur plusieurs configurations tel que la commande de quatre ponts de Graetz indépendants, la commande d'un redresseur dodécaphasé et la commande d'un redresseur à vingt-quatre phases.

Cependant il est possible de perfectionner le système de commande. Ainsi, en mesurant les courants de charge avec le micro-contrôleur maître nous pourrions assurer la protection rapide de courant de tous les redresseurs commandés. En plus, la fréquence de la mise à jour des données affichées à l'écran de l'ordinateur peut être augmentée en transmettant périodiquement au micro-ordinateur toute la zone de mémoire de la matrice « état » sous forme d'un paquet de données. L'ordinateur va identifier les données en fonction de leur position dans le paquet de données reçu. Finalement, le temps de réponse de la boucle de réglage de tension peut être diminué en remplaçant le régulateur P de la boucle de tension par un régulateur PID.

La carte d'interface développée et le logiciel conçu représentent des outils de base pour l'évaluation expérimentale de nouvelles configurations de convertisseurs développés à la Section Électrotechnique de l'École Polytechnique de Montréal.

BIBLIOGRAPHIE

1. LÉONARD, E. (1993). Multicontrôleur Forth pour les applications en temps réel, Mémoire de maîtrise, École Polytechnique de Montréal, Canada.
2. OLIVIER, G., GAGNÉ, M. et BEAUDRY, S. (1989). Convertisseur quadri-hexaphasé 5000 V / 300 A, Rapport technique, École Polytechnique de Montréal, Canada.
3. CYRIL W. LANDER (1987), Power Electronics, Leicester Politechnic, McGraw-Hill.
4. OLIVIER, G. (1982). Études comparative et commande de convertisseurs triphasés avec applications aux machines électriques. Thèse de doctorat, Université Concordia, Canada.
5. LÉONARD, E. et APRIL, G.-E. (1994). Compilateur Forth à chaînage direct pour le microcontrôleur M37700. Rapport technique, École Polytechnique de Montréal, Canada.
6. LÉONARD, E. et APRIL, G.-E. (1994). Compilateur Forth à chaînage par sous-routines pour le microcontrôleur M37450. Rapport technique, École Polytechnique de Montréal, Canada.

ANNEXE A

Le tableau A.1 présente la matrice de variables réelles « état ». On identifie dans ce tableau les valeurs de la variable de type « Off... » qui donne la position dans la matrice « état » d'un groupe de données. La variable « pont » prend toujours des valeurs entre 0 et 4. Le même tableau présente la matrice « trans » pour souligner la correspondance entre les données de la matrice « état » et leurs drapeaux enregistrés dans la matrice « trans ». La matrice « état » contient aussi le code des messages qui sont envoyés à l'ordinateur.

Tableau A.1: Contenu de la matrice de variables réelles "état".

Matrice des réels <i>état</i>						Matrice <i>trans</i>
Indice		Variable off_xx	Valeur de off_xx	Pont	Information à transmettre	1 = nouvelle donnée 0 = donnée transmise
dec	hex					
1	1	Off_Us	1	0	Tension du pont 0	
2	2			1	Tension du pont 1	
3	3			2	Tension du pont 2	
4	4			3	Tension du pont 3	
5	5			4	Tension moyenne	
6	6	Off_Is	6	0	Courant du pont 0	
7	7			1	Courant du pont 1	
8	8			2	Courant du pont 2	
9	9			3	Courant du pont 3	
10	A			4	Courant moyen	

Matrice des réels <i>etat</i>						Matrice <i>trans</i>
Indice		Variable off_xx	Valeur de off_xx	Pont	Information à transmettre	1 = nouvelle donnée 0 = donnée transmise
dec	hex					
11	B	Off_C	B	0	Consigne du pont 0	
12	C			1	Consigne du pont 1	
13	D			2	Consigne du pont 2	
14	E			3	Consigne du pont 3	
15	F			4	Consigne générale	
16	10	Off_Isa	10	0	Courant ancien du pont 0	
17	11			1	Courant ancien du pont 1	
18	12			2	Courant ancien du pont 2	
19	13			3	Courant ancien du pont 3	
20	14			4	Courant moyen ancien	
21	15	Off_Z	15	0	Impédance du pont 0	
22	16			1	Impédance du pont 1	
23	17			2	Impédance du pont 2	
24	18			3	Impédance du pont 3	
25	19			4	Impédance générale	
26	1A	Off_pt	1A	0	Point de travail du pont 0	
27	1B			1	Point de travail du pont 1	
28	1C			2	Point de travail du pont 2	
29	1D			3	Point de travail du pont 3	
30	1E			4	Point de travail équivalent	
31	1F	-	-	-	Courant du maître	
32	20	-	-	-	Tension du réseau	
33	21	Off_UI	21	0	Tension de ligne du pont 0	
34	22			1	Tension de ligne du pont 1	
35	23			2	Tension de ligne du pont 2	

Matrice des réels <i>etat</i>						Matrice <i>trans</i>
Indice		Variable off_xx	Valeur de off_xx	Pont	Information à transmettre	1 = nouvelle donnée 0 = donnée transmise
dec	hex					
36	24	Off_UI	21	3	Tension de ligne du pont 3	
37	25			4	Tension de ligne du pont équivalent	
38	26	-	-	-	Tension du réseau grande	
39	27	-	-	-	Tension du réseau petite	
40	28	-	-	-	Tension du réseau normale	
41	29	-	-	-	Courant des esclaves normal	
42	2A	-	-	-	Courant des esclaves anormal	
43	2B	-	-	-	Surcharge esclaves de longue durée	
44	2C	-	-	-	Courant du maître normal	
45	2D	-	-	-	Courant du maître normal	
46	2E	-	-	-	Courant du maître trop grand	
47	2F	-	-	-	Surcharge du maître de longue durée	

ANNEXE B

Le tableau B.1 montre la matrice d'entiers « b_etat ». Sa structure est similaire à la matrice « etat » présentée dans l'annexe A.

Tableau B.1: Contenu de la matrice d'entiers "b_etat"

Indice		Variable Off_xx	Valeur de Off_xx	Valeur de la variable <i>pont</i>	Nature des données
dec	hex				
1	1	Off_Us	1	0	Tension de sortie du pont 0
2	2			1	Tension de sortie du pont 1
3	3			2	Tension de sortie du pont 2
4	4			3	Tension de sortie du pont 3
5	5			4	
6	6	Off_Is	6	0	Courant de sortie du pont 0
7	7			1	Courant de sortie du pont 1
8	8			2	Courant de sortie du pont 2
9	9			3	Courant de sortie du pont 3
10	A			4	
11	B	Off_C	B	0	Consigne du pont 0
12	C			1	Consigne du pont 1
13	D			2	Consigne du pont 2
14	E			3	Consigne du pont 3
15	F			4	
16	10	Off_Isa	10	0	Courant ancien du pont 0
17	11			1	Courant ancien du pont 1
18	12			2	Courant ancien du pont 2
19	13			3	Courant ancien du pont 3

Indice		Variable Off_xx	Valeur de Off_xx	Valeur de la variable pont	Nature des données
dec	hex				
20	14	Off_lsa	10	4	Courant ancien moyen
21	15	Off_sis	15	0	Signe du courant du pont 0
22	16			1	Signe du courant du pont 1
23	17			2	Signe du courant du pont 2
24	18			3	Signe du courant du pont 3
25	19	Off_sus	1A	0	Signe de la tension du pont 0
26	1A			1	Signe de la tension du pont 1
27	1B			2	Signe de la tension du pont 2
28	1C			3	Signe de la tension du pont 3
29	1D	Off_Usa	1D	0	Tension de sortie ancienne 0
30	1E			1	Tension de sortie ancienne 1
31	1F			2	Tension de sortie ancienne 2
32	20			3	Tension de sortie ancienne 3
33	21	Off_Ca	21	0	Consigne ancienne du pont 0
34	22			1	Consigne ancienne du pont 1
35	23			2	Consigne ancienne du pont 2
36	24			3	Consigne ancienne du pont 3
37	25	Off_la	25	0	Courant ancien du pont 0
38	26			1	Courant ancien du pont 1
39	27			2	Courant ancien du pont 2
40	28			3	Courant ancien du pont 3

ANNEXE C

Des nouveaux mots Forth ont été définis pour faciliter l'accès aux matrices « état » et « b_etat ».

send (donnée réelle, indice matrice -- ...

Ce mot reçoit la donnée réelle et l'indice de la matrice « état ». La donnée est écrite dans la matrice « état » à la position donnée par l'indice.

bsend (donnée 8 bits, indice matrice --...

Action similaire à send. Un entier de 8 bits est écrit dans la matrice des entiers b_etat à la position donnée par « index ».

pfsend (donnée réelle, indice --...

Écrit la donnée dans la matrice « état » à la position donnée par la somme de l'indice de la matrice « état » et le numéro du pont courant.

pbsend (donnée réelle, indice matrice -- ...

Similaire à pfsend. Un entier de 8 bits est écrit dans la matrice des entiers b_etat à la position donnée par la somme de l'indice de la matrice « b_etat » et le numéro du pont courant.

pget (indice matrice -- entier 16 bits

Récupère de la matrice « b_etat » une donnée entière de 16 bits qui se trouve à la position donnée par la somme de l'indice de la matrice « _etat » et le numéro du pont courant.

pbget (indice matrice -- entier 8 bits

Récupère de la position (indice + numéro du pont) de la matrice *b_etat* un entier de 8 bits qui se trouve à la position donnée par la somme de l'indice de la matrice « *b_etat* » et le numéro du pont courant.

pfget (indice matrice -- donnée réelle

Récupère de la matrice «*état*» une donnée réelle qui se trouve à la position donnée par la somme de l'indice de la matrice « *état* » et le numéro du pont courant.

ANNEXE D

L'annexe D présente le logiciel des micro-contrôleurs esclaves écrit en Forth et Assembleur. Le programme réalise la commande des thyristors, la mesure de la tension, du courant de sortie de l'angle de commande et de la période de la tension du réseau. On transmet au micro-ordinateur ce programme en premier en utilisant les logiciels Procom ou Terminal de Windows.

```
( *
( *      SYSTÈME DE COMMANDE DES REDRESSEURS      *
( *
( *      PROGRAMME DE COMMANDE DU MICRO-CONTRÔLEUR ESCLAVE *
( *
( *
( *
( *
( * Version initiale : 1.0 Adrian-Valentin Fadei      26 -août- 1995      *
( *      Email: valentin@electech.polymtl.ca
( *      fadei@cae.ca

( programme des micro-contrôleurs esclaves qui commandent chacun un pont de Graetz
  0 | ( selection du mP esclave 0 pour le chargement
    ( de la memoire.

    hex
    0 E8 b! ( désactive la communication sériele

    F0 constant T1L ( registre de la minuterie 1
    F1 constant T1H
    F2 constant TL1L ( «latch « de la minuterie 1.
    F3 constant TL1H
    ED constant T1C ( registre de contrôle de la minuterie 1
    F4 constant T2L ( registre de la minuterie 2
    F5 constant T2H
    F6 constant TL2L ( «latch « de la minuterie 2.
    F7 constant TL2H
    EE constant T2C ( registre de contrôle de la minuterie 2
    F8 constant T3L ( registre de la minuterie 3
    F9 constant T3H
    FA constant TL3L ( «latch « de la minuterie 3.
    FB constant TL3H
    EF constant T3C ( registre de contrôle de la minuterie 3
    FE constant IC1 ( registre de contrôle des interruptions 1
    FF constant IC2 ( registre de contrôle des interruptions 2
    E2 constant Ad ( registre du convertisseur A/D
    E3 constant AdC ( registre de contrôle du convertisseur A/D
    DE constant MIS1 ( registre divers 1
    D6 constant P3 ( port 3
```

```

D7 constant P3d
D8 constant P4
DC constant P6
DD constant P6d
1 variable Rnl 14 Rnl b!
1 variable Rnh 20 Rnh b!
1 variable RI 14 RI b!
1 variable Rh 20 Rh b!
1 variable angle_mesurel 14 angle_mesurel b!
( angle mesuré, valeur initiale pour 90gr
1 variable angle_mesureh 29 angle_mesureh b!

( *****
1 variable fangle_mesurel 14 fangle_mesurel b!
( angle mesuré, valeur initiale pour 90gr
1 variable fangle_mesureh 29 fangle_mesureh b!
1 variable cmdTh 21 cmdTh b!
1 variable step 1 step b!
programme de réglage [0-5]
1 variable ind 0 ind b!
1 variable tens 0 tens b!
1 variable temp 0 temp b!
1 variable tens_old 0 tens_old b!
1 variable temp_old 0 temp_old b!
1 variable s_tens 0 s_tens b!
1 variable s_temp 0 s_temp b!
1 variable mes 1 mes b!
2 variable xf 0 xf !
2 variable yf 0 yf !
1 variable tempf
1 variable tensf
1 variable 60grl 20 60grl b!
1 variable 60grh 1B 60grh b!
( *****
1 variable f60grl 20 f60grl b!
1 variable f60grh 1B f60grh b!
4 variable Tau 4 1A04 Tau d!
4 variable Tau2 4 1A04 Tau2 d!
4 variable Delta 0 0 Delta d!
1 variable tauI A6 tauI b!
1 variable tauh A2 tauh b!
1 variable tau2I 22 tau2I b!
1 variable tau2h 22 tau2h b!
1 variable anglI 20 anglI b!

1 variable anglh 1B anglh b!
1 variable compl 0 compl b!
1 variable comph 0 comph b!
1 variable ttour 0 ttour b!
1 variable tour 0 tour b!
1 variable fm 1 fm b!

1 variable fs 1 fs b!
hex
1 variable 15grl 20 15grl b!

1 variable 15grh A 15grh b!
1 variable -45grl 00 -45grl b!

```

(registre de direction du port 3
(port 4
(port 6
(registre de direction du port 3
(angle reçu du maître
(angle reçu du maître

(mot de commande du pont 33dec=21hex
(numéro de l'itération faite par le

(tension de sortie du pont
(courant du pont
(tension ancienne
(courant ancien
(signe de la tension de sortie
(signe de la tension de sortie

(délais initial d'allumage

(délais initial d'allumage

(Valeur moyenne de la période * 256

(valeur de la période mesure par TB0
(16662dec = 4116hex

(délai entre les commandes des th. 60
deg. comp

(compensation du délais

(drapeau qui indique la nature de la
mesure

(angle fixe de 15 degrés -> délai minimal
(entre les commandes des thyristors.

1 variable -45grh -11 -45grh bl

(angle fixe de 45 degrés -> correction
(maximale du délai entre les commandes
(des thyristors [s'applique pour la
(diminution de l'angle de commande].

1 variable 165grl 00 165grl bl
1 variable 165grh 4B 165grh bl

(angle fixe de 165 degrés -> correction
(maximale du délai entre les commandes
(des thyristors [s'applique pour
(l'augmentation de l'angle de commande].

1 variable c0 0 c0 bl
1 variable c1 88 c1 bl
1 variable c2 1F c2 bl
1 variable b0 0 b0 bl
1 variable b1 A2 b1 bl
1 variable b2 A5 b2 bl
2 variable ind

(adresse du table qui contient les mots de
(commande des thyristors.

2 variable bbase
2 variable lind 0 lind l
1 variable count 1 count bl
(1 , 2 , 4 , 8 , 16 , 32 , commande courte il faut ajouter la mise à zéro des th après les commandes
(commande de a+ c- b+ a- c+ b- pour synchronisation. sur Uca
(33 , 3 , 6 , 12 , 24 , 48 , commande longue de 120 degrés.
(il faut enlever de TA4Int le FF P7 bl
hex
here ind l
6 allot

(dec pour 1 8 16 32 64 128 commande sur H

82 ind @ 0 + bl

(02 04 08 20 40 80 60 degrés

06 ind @ 1 + bl

0C ind @ 2 + bl

28 ind @ 3 + bl

(82 06 0C 28 60 C0 120 degrés

60 ind @ 4 + bl

C0 ind @ 5 + bl

1 IC1 bl (interdire les interruptions

0 IC 2 bl (interdire les interruptions

0 P3d bl (faire P3 port d'entrée

0 P6d bl (faire P6 port d'entrée

(programme pour le calcul des intervalles entre les commandes des thyristors

(pour les esclaves

(reçoit la consigne du délai sur la pile

code cinterv (n -- ...

FE lda,

(interdiction de l'interruption inbuf

FE # and,

FE sta,

clc,

Rh lda,

(la consigne va rester low->A high-

>Y

rora,

tay,

clc,


```

15grl lda, ( 15gr -> 694.44435microsecondes
TL1L sta, ( ici on commande le délai minimal
15grh lda, ( entre les commandes des thyristors 15 degrés électriques
TL1H sta,
else, ( la correction est plus petite que 45 degrés
    tau2l lda, (
    compl sta,
    tau2h lda,
    comph sta,
    tau2l lda, ( 60 degrés - correction = nouveau
                ( délai entre les commandes des thyristors

    clc,
    f60grl adc,
    TL1L sta,
    tau2h lda,
    f60grh adc,
    TL1H sta,
    then,
    then,
    fangle_mesurel lda,
    fangle_mesureh ldy,
    clc,
    compl adc,
    fangle_mesurel sta,
    tya,
    comph adc,
    fangle_mesureh sta,
    0 # lda, ( 0 comp !
    compl sta,
    comph sta,
    ( Rnl lda,
    ( Rl sta,
    ( Rnh lda,
    ( Rh sta,
    FE lda,
    1 # ora,
    FE sta,
    rts,
end-code
( extrait la prochaine commande qui sera envoyé aux thyristors
: cmdthy ( ..._...
step b@ dup ( step indique l'itération qui suit
ind @ + b@ cmdTh bl ( Obtient la commande correspondante à l'itération qui suit
1+ dup ( variable step sur la pile
( .....
( mesure de la tension de sortie sans échantillonnage
( section utile si on désire la commande échantillonnée de la tension de sortie
    dup
    6 =
    if
        P6 b@ 1 or P6 bl
    then
( .....
    6 =
    if
        drop 0 ( on initialise séquence de commande
                ( le prochain mot de la séquence de
                ( commande sera le premier « wrap around »
                ( mot défini dans la matrice cmdTh
    then

```

```

                                step b!
;
1  variable mes 1 mes b!
2  variable xf 0 xf !
2  variable yf 0 yf !
1  variable tens
1  variable tempf                ( valeur filtrée du courant de sortie
1  variable tensf                ( valeur filtrée de la tension de sortie
1  variable filtf 2 filtf b!      ( constante du filtre numérique = minimum 2
1  variable tau 4F tau b!

: ftens ( ... -- ...
xf @ + xf 1 + b@ - xf !
xf 1 + b@ tensf b! ;           ( filtre pour la tension

i: InBuf ( ...
  ( ** interruption pour recevoir les consignes du maître
  ( E5 adresse de Data bus buffer status register
  ( E4 adresse de Data bus buffer register == comm. avec le maître
  ( ** Début/Arrêt des commandes ou de la transmission
code
E5 lda,
8 # and,
eq if,                          ( vérifie si au moment de la transmission le bit
    E4 lda,                      ( A0 est 1 => bit low
    RI sta,
else,                            ( si c'est un bit H. alors il vérifie si ce n'est pas
    E4 lda,                      ( un mot de commande
    FF # cmp,                   ( si le mot FF est reçu il arrête les commandes aux thyristors
    eq if,
    0 # lda,
    fs sta,
else,
    FE # cmp,                   ( si le mot FE est reçu il active les commandes des th
    eq if,
    0 # lda,
    tour sta,
    1 # lda,
    fs sta,
else,
    FB # cmp,                   ( si le mot FB est reçu il arrête la transmission périodique des données
    eq if,
    0 # lda,
    count sta,
else,
    FA # cmp,                   ( si FA est reçu il force le mP de rapporter
    immédiatement les mesures
    eq if,                      ( et déclenche la transmission périodique des données
        F1 # lda,
        count sta,
    else,
        165grh cmp,
        pl if,                  ( limite l'angle maximal reçu
        165grh lda,
        then,
        1 # cmp,                ( limite l'angle minimal reçu
        mi if,
        1 # lda,

```

```

                                then,
                                Rh sta,
                                then,
                                then,
                                then,
                                then,
                                Forth
                                ;i
1  variable mesuré 0 mesuré b!

( ** démarrage de la mesure de la tension de sortie
i: int2 ( ..._...
    code
    ( 2 P3 seb,
    ( ** déclenche la mesure
    ( 4 P3 clb,
    ( Adc est le registre de contrôle du A/D

    ( ** début de la mesure de tension
    ( choix du canal 0
    ( 0 # lda,
    ( AdC sta,
    ( 2 P3 clb,

    ( ** chien de garde pour la réception et transmission des données par l'esclave
    ( E5 lda,
    ( 1 # and,
    ( eq if,
    ( 2 DC clb,
    ( else,
    ( 2 DC seb,
    ( then,
    ( E5 b@ 1 and 0 =
    ( if
    ( else
    ( 4 DC b@ or DC b!
    ( FB DC b@ and DC b!
    ( then
    Forth
    ;i

( filtre numérique pour le courant mesuré
( seulement les petites variations sont filtrées
( le filtre ne s'applique pas aux grandes variations
: tempfilt ( ..._ n
    temp b@ tempf b@ - abs filt b@ 3 - <
    if
        yf @ dup tau b@ / -
        tempf b@ + dup yf !
        tau b@ / tempf b!
    else
        temp b@ dup tempf b!
        tau b@ * yf !
    then
    tempf b@ temp_old b@ - abs
    ;
( filtre numérique pour la tension mesurée
( seulement les petites variations sont filtrées

```

(si ces ne sont pas des commandes alors c'est un
(byte H. de la consigne

(interruption synchrone au signal extérieur INT2

(Adc est le registre de contrôle du A/D

(vérification du registre de sortie

(si plein transmet l'interruption

(si vide efface la demande d'interruption

retourne la valeur filtrée

(il passe par le filtre pour des petites variations

(pour des grandes variations il met à jour
(le filtre et il transmet tel quel la donnée

(laisse sur la pile le résultat du filtre

```

( le filtre ne s'applique pas aux grands variations
: tensfilt ( ... --- n
    ( filtre pour la tension de sortie
    tens b@ tensf b@ - abs filt b@ <
    if
        ( il passe par le filtre pour des petites variations
        xf @ dup tau b@ / -
        tens b@ + dup xf !
        tau b@ / tensf b!
    else
        ( pour des grandes variations il met à jour
        ( le filtre et il transmet tel quel la donnée
        tens b@ dup tensf b!
        tau b@ * xf !
    then
        tensf b@ tens_old b@ - abs
;
1 variable logic 0 logic b!
: transmit ( ...---...
    ( ** programme dans le « background » pour transmission
    ( fm drapeau de tour de role u=0/t=1
    ( E5 adresse de Data bus status register
    ( DC adresse du registre P6
    ( des donnees au maître
    ( tension sortie et courant
    ( ** filtrage de la tension mesure

begin
( 4 DC b@ or DC b!
E5 b@ 1 and
0 =
attend
if
    ( efface toute demande d'interruption
    ( vérifie si le buffer de sortie
    ( si il n'est pas vide il ne fait rien et il
    ( qu'il soit libre
    4 DC b@ or DC b!
    (
    E5 b@ FB and E5 b!

    fm b@ dup 1 =
    tension / courant
    if
        tensfilt 0 fm b!
    else
        tempfilt 1 fm b!
    then
    ( af+
    count b@ 0 <>
    if
        ( si count =0 ne transmet pas périodiquement
        ( si count <> de 0 continue la transmission
        ( incrémente le conteur pour l'envoi des données
        ( f1 doit être impaire
        ( si 34 hex cycles sont passées, il envoie la
        count b@ FE >
        if
            donnée
            drop 10
            1 count b!
            then
            ( initialise count
            then
            ( af-
            3 >
            if
                ( si la différence est plus grande que 1 on transmet
                ( annule l'interruption input buffer full
                ( s' il s'agit de la tension
                ( mettre bit 6 à 0 pour tension
                FE b@ FE and FE b!
                1 =
                if
                    E5 b@ BF and E5 b!

```

```

        tensf b@
        BusOut
FB DC b@ and DC b!
tensf b@ tens_old b!

        else
            E5 b@ 40 or E5 b!
            tempf b@
            BusOut
FB DC b@ and DC b!
tempf b@ temp_old b!
        then
            (
                4 DC b@ or DC b!
            (
                FE b@ 01 or FE b!
            else
                drop
            then
            else
            ( FB DC b@ and DC b!
            ( 4 DC b@ or DC b!
            then
            again
        ;
i: ADInt (...
( apparaît a la fin d'une conversion A/D
( ** mesure la tension et le courant
code
( 2 P3 seb,
ttour lda,
mi lf,
        Ad lda,
        tens sta,
        P4 lda,
        clt,
        4 # and,
        eq lf,
            80 # lda,
            s_tens sta,
        else,
            0 # lda,
            s_tens sta,
        then,
            (
                0 P6 clb,
fin mesure
                1 # lda,
sortie
                AdC sta,
                1 # lda,
                ttour sta,
            else,
                Ad lda,
                temp sta,
                -1 # lda,
                ttour sta,
            then,
            (
                2 P3 clb,
Forth
;i

```

(interruption transmise vers le maître

(envoie à la sortie la tension

(mettre bit 6 à 1 pour le courant

(interruption transmise vers le maître

(fin de l'impulsion d'interruption

(permet les interruptions input buffer

(interruption transmise vers le maître

(fin de l'impulsion d'interruption

(vérifie à qui est le tour tension/courant

(lecture de la tension P4.0

(lecture du signe

(vérification du signe

(positif

(négatif

(mise en conduction du mosfet {set pour 474 ->

(fin intégration de la mesure de la tension de

(start de l'acquisition du courant

(lecture du courant

(met le drapeau pour la mesure de tension

```

( mesure de la période
i: Ev2 ( ...
( P3 b@ DF and P3 b!
code
    0 # lda,
    FE sta,

( 2 P3 seb,
Forth
FE b@ FE and FE b!
tour b@ 0 = if
( 29 IC1 b!
( 26 IC2 b!
( 24 T3C b!
( 23 T3C b!
( 2 T1C b!
    46 TL1L b!
    41 TL1H b!
    10 MIS1 b@ or MIS1 b!
1 tour b!
then
code
(
    4 P3 clb,
sec,
b0 lda,
b1 sbc,
b0 sta,
    b1 lda,
    b2 sbc,
    b1 sta,
b2 lda,
0 # sbc,
b2 sta,
    clc,
b0 lda,
TL2L adc,
b0 sta,
    b1 lda,
    TL2H adc,
    b1 sta,
    taul sta,

b2 lda,
0 # adc,
b2 sta,
tauh sta,
clc,
( clt,
( 5 P3 seb,
clc,
tauh lda,
rora,
tauh sta,
taul lda,
rora,
taul sta,
29 # lda,
FE sta,
Forth

```

(interdit les interruptions
(il s'exécute seulement au démarrage

(X - x/256 filtre pour la période mesuré

(entrée du filtre: byte moins important

(entrée du filtre: byte plus important

(sortie du filtre: byte moins important

(sortie du filtre: byte plus important
(fin filtre


```

taul @ 3 / 10 + 60gr!      ( calcule 1/6 période
FE b@ 01 or FE bl          ( permet les interruptions
;

(  mesure de l'angle de commande
i: Ev3 ( ..._...
code
(  2 P3 seb,
    00 # lda,
    FE sta,
    20 # lda,
    FF sta,
    cll,
    TL3L lda,              ( lit dans T3L et T3H l'angle mesure
    TL3H ldy,
    clc,
    80 # adc,
    fangle_mesurel sta,
    tya,
    0 # adc,
    fangle_mesureh sta,
    FF # lda,
    compl sta,             ( remet a zéro la correction
    comph sta,             ( 0 comp !
(  2 P3 clb,
    29 # lda,
    FE sta,
    26 # lda,
    FF sta,
    Forth
;

i: IntT1 ( ..._...
( programme principal
( ** calcule le délai entre les commandes ---> cinterv
( ** choisit et transmet les commandes ---> cmdthy
( ** peut envoyer une commande aveugle -- seulement th.1 pour synchroniser l'esclave
code
step lda,
5 # cmp,
eq if,
    0 # lda,              ( lecture de la tension de sortie
    AdC sta,
then,
Forth
FE b@ FE and FE bl        ( interdiction de l'interruption inbuf reg. à l'adresse FE
cinterv                  ( calcul du prochain délai
FE b@ 01 or FE bl        ( accepte les interruptions de input buffer
cmdthy                   ( choix du mot de la prochaine commande
(  3 P3 seb,
code
fs lda,
1 # cmp,
eq if,
    cmdTh lda,            ( transmission séquentielle
    E0 # and,             ( des trois premiers bits du mot de commande.
    P3 sta,

```

```

( signal pour enregistrer sur le front croissant dans le premier
registre
3 P3 seb,
3 P3 clb,
cmdTh lda,
(
(
( transmission séquentielle des trois derniers bits du mot de
commande.
asia, ( 16*
asia,
asia,
asia,
E0 # and,
P3 sta,
( sortie sur le port
( signal pour enregistrer dans le deuxième registre
( sur le front croissant
4 P3 seb,
4 P3 clb,
(
else,
( ***** transmission aveugle des commandes aux th.
( seulement th.1 reçoit la commande
( cela nous permet de garder la synchronisation du pont même
( si on a commandé
( l'arrêt de la commande des thyristors.
( transmission des trois premiers bits du mot de commande.
cmdTh lda,
E0 # and,
30 # and,
P3 sta,
( signal pour activer le premier registre
( sur le front croissant
3 P3 seb,
3 P3 clb,
(
0 # lda,
P3 sta,
( avoir « high » à la sortie s'assurer que les th. sont éteints
( signal pour activer le deuxième registre
( sur le front croissant
4 P3 seb,
4 P3 clb,
(
then,
0 P6 seb,
( fin du signal transmis au 474
(
Forth
1 count +!
60grl b@ f60grl bl
60grh b@ f60grh bl
;)

( on enregistre les sous_routines pour les demandes d'interruption
: setVect ( ..._...
' Int2 FFF6 !
' ADint FFE2 !
' Ev2 FFEA !
' Ev3 FFE8 !
' IntT1 FFF2 !
' InBuf FFFC !
;

( on active les interruption "enable interrupts"
: setEnable ( ..._...
29 IC1 bl ( 29 t1, int2, bufIn
26 IC2 bl ( ad, ev2, ev3
;

: setPort ( ..._...
FD P6d bl ( 0 cmd tr, 1 int2, 2 intr. vers maître

```

```

    F9 P3d b!
;

:   setMode ( ..._...
    0 E8 b!          ( annule la communication sériele de l'esclave
                    ( pour libérer les ports
    24 T3C b!        ( Pulse width measurement mode
    23 T2C b!        ( pulse period measurement
    2 T1C b!         ( Pulse output mode
;

(   on met en route les minuterics
:   setStart ( ..._...
    MIS1 b@ 60 or MIS1 b!   ( t3,t2-start
;

:   s ( ..._...
    hex
    setMode
    setVect
    setPort
    setEnable
    MIS1 b@ 60 or MIS1 b!   ( t3,t2-start
bye
;

(   mise en route de l'esclave
:   l ( ..._...
s   transmit
;
:   set ( ..._...
    hex
    setMode
    setVect
    setPort
    setEnable
    setStart
bye
;

(   ' l ustart !          ( démarrage du premier coup
(   9C00 bbase !
UpdTb!
bye
0>123
(   on utilise les mots suivants si on désire
(   faire fonctionner l'esclave
(   tout seul sans le maître
(   on fait:
(   no. esclave | <cr>
(   l <cr>
(   testesc <cr>
(   la consigne se transmet avec:
(   consigne no.pont >cp' <cr> pour byte hight
(   consigne no. pont >cp <cr> pour le byte low
(   permet la commande des thyristors
:   qw ( ..._...
FF P5d b!

```

```
;
: testesc ( ..._...
44 TA4m bl
28 TA4 l
3 TA4lcr bl
4 CntSt #orb!
qw
;
```

(mise en route de l'horloge de 25 Khz
(pour l'alimentation des AOP AD204.

34 1 1 mat b_etat	(matrice pour conserver l'état du système
9 2 1 mat _etat	(matrice pour conserver les consignes aux
	(ponts -entiers double
37 6 1 mat etat	(matrice pour conserver l'etat du système
34 1 1 mat tran	(pour la transmission au P.C. - réels
(Offset commun pour etat et b_etat	
1 variable Off_Us	(tension de sortie
1 variable Off_Is	(courant de sortie
1 variable Off_C	(consigne du P.C.
1 variable Off_Isa	(courant ancien de sortie
(Offset seulement pour la matrice etat	
1 variable Off_pt	(point travail
1 variable Off_UI	(tension de ligne appliquée au
	(pont de Graetz
(offset seulement pour la matrice b_etat	
1 variable Off_sis	(signe courant des esclaves
1 variable Off_sus	(signe tension de sortie
1 variable Off_Usa	(tension sortie ancienne
1 variable Off_Ca	(consigne ancienne
1 variable Off_t	(point travail
(offset pour la matrice _etat	
1 variable Off_R	(consigne au pont
1 variable Off_dR	(correction des consignes au pont
(*** coefficients de proportionnalité	
5 variable kUc 1, kUc f!	(consigne
5 variable kUr 1, kUr f!	(tension du réseau
5 variable kIs 1, kIs f!	(courant de sortie
5 variable kUs 1, kUs f!	(tension de sortie
5 variable kIm 1, kIm f!	(courant du maître
1 variable pont	(indique le pont qui est réglé pendant
	(l'itération en cours
1 variable idx 0 idx b!	
1 variable req 0 req b!	(registre qui contient les ponts
	(qui doivent être réglés
1 variable vpar 1 vpar b!	(drapeau de variation
1 variable variation	(indique si il y a eu une variation
	(des paramètres extérieurs
(_____ Variables concernant le MAÎTRE	
1 variable Ur 0 Ur b!	(tension du réseau
1 variable Ura 0 Ura b!	(tension réseau ancienne
1 variable Su 0 Su b!	(signe tension réseau
1 variable bUrmax	(limite maximale de la tension du réseau
1 variable bUrmin	(limite minimale de la tension du réseau
5 variable Urmax	(limite maximale de la tension du réseau
5 variable Urmin	(limite minimale de la tension du réseau
5 variable tens_reseau	(tension du réseau courante
5 variable Ucmx	(consigne de tension maximale
1 variable Im 0 Im b!	(courant de sortie
1 variable Ima 0 Ima b!	(courant ancien de sortie
1 variable bImnom	(courant nominal du maître
1 variable bImmx	(courant maximal admissible
1 variable Si 0 Si b!	(signe du courant du maître
5 variable Imn	(courant nominal du maître
5 variable Immx	(courant maximal du maître

```

( _____ Variables concernant les ESCLAVES _____
1 variable bUsmax 0 bUsmax b!          ( tension de max. sortie en 8 bits
5 variable Usca!                        ( tension pour laquelle est calibre la sortie
5 variable Usmax                        ( tension maximale a la sortie
                                       ( coïncide avec la tension maximale de la
consigne Ucmax 5 variable Isn          ( courant nominal d'un esclave
5 variable Ismax                       ( courant maximal d'un esclave
1 variable blsnom                      ( courant nominal d'un esclave
1 variable blsmax                      ( courant maximal d'un esclave
1 variable cled 0 cled b!
1 variable cpont 0 cpont b!
1 variable kc
1 variable al
1 variable c1led
1 variable c1
1 variable k1c
1 variable a1l
1 variable ks 1 ks b!                  ( sensibilité aux variations
1 variable k 0 k b!                   ( variable
1 variable r 0 r b!                   ( variable
1 variable r1 0 r1 b!
2 variable Umoy 0 Umoy !               ( tension de sortie moyenne
1 variable pontc                       ( pont pour lequel on vérifie le courant de
sortie
1 variable step FF step b!             ( durée maximale du courant de surcharge
pour le maître
1 variable sstep FF sstep b!          ( durée maximale du courant de surcharge
pour les esclaves
5 variable Ud 2, Ud f!
( ***** variables initiales
2 variable Un 0 Un !                   ( tension nominale du réseau
5 variable x 0,0 x f!
2 variable index 0 index !
2 variable h_index 0 h_index !
2 variable index_h 0 index_h !
2 variable index_hs 0 index_hs !
2 variable v_index 0 v_index !
2 variable Rp 0 Rp !
2 variable Rs 0 Rs !
1 variable mask B mask b!

( ***** matrice qui contient les variables du système
7F 20 2 2 mat comm

( _____
( définition des mots d'accès aux matrices etat et b_etat

: send ( f, b --- ...
( reçoit sur la pile la donnée réelle et la position dans la matrice ( initialise le drapeau: nouvelle donnée.
( utilisation: float , position dans le tableau, send
etat f! 1 vpar b!
;
: bsend ( b1, b2 --- ...
( enregistre une donnée dans la matrice b_etat
( reçoit sur la pile la donnée,e et la position dans la matrice
( , , bsend
b_etat b!

```

```

;
pfsend ( f, ad --- ....
    ( enregistre une donnée réelle dans la matrice r,elle
    ( a la position correspondant au pont courant
    ( reçoit sur la pile la donnée,e et l'offset dans la matrice(
    ( float offset pfsend
b@ pont b@ + dup etat f!
1 swap tran b! 1 vpar b!
;
: p_send ( d, ad --- ...
    ( enregistre une donne entière double
    ( dans la matrice d'entiers doubles
    ( a la position correspondant au pont courant
    ( met a un le drapeau nouvelle donnée
    ( reçoit sur la pile la donnée,e et l'offset ( donnée offset p_send
b@ pont b@ + _etat !
1 vpar b!
;
: pbsend ( b, ad --- ...
    ( envois dans la matrice d'entiers au pont courant
    ( reçoit sur la pile la donnée,e et l'offset
    ( valeur offset pbsend
b@ pont b@ + b_etat b!
1 vpar b!
;
: fget      ( ad --- f
    ( récupère de la matrice une donnée,e r,elle pour le pont courant
    ( offset fget
b@ pont b@ + etat f@
;
: bget      ( ad --- b
    ( récupère de la matrice une donnée entière pour le pont courant
b@ pont b@ + b_etat b@
;
: gett      ( ad --- d
    ( récupère de la matrice une donnée entière double pour le pont courant
b@ pont b@ + _etat @
;
2 variable R 0 R !
(
-----
( ***** Sous-programmes de commande du mC MAÎTRE *****
: stop_procedure      ( ...---...
    ( ** commande d'extinction des ponts
    (
    0 CntSt b!
    3 setlevel
    FF P5 b!
." arrêt du système " cr
;
: clln ( n --- ...
    ( reçoit sur la pile le no. du led a allumer: 2-7
    ( met a un le bit correspondant au pont requérant
    ( mise en route seulement si il ne l'était pas
cled #orb!
CntSt b@ 8 and 0 =
if
    0 al b! ( drapeau de on off
    3 CntSt #orb!
    ( mise en route de l'horloge
then
;

```



```

: Resc ( b ---...
    ( reçoit dans la pile: no. du pont
    ( traite les interruption d'un esclave
    ( indique par le nombre reçu sur la pile
    ( pile vide 'a la sortie
    dup dup ?cp dup dup
    0001 and 0<
    if
    0040 and iof 0<
    donnée
    if
    0080 and 0<
    if
    Off_sis b@ + 0 swap bsend
    matrice
    else
    Off_sis b@ + 1 swap bsend
    then
    dup cp> swap
    Off_ls b@ + b_etat b!
    else 80 and 0<
    if
    Off_sus b@ + 0 swap bsend
    matrice
    else
    Off_sus b@ + 1 swap bsend
    then
    dup cp> swap
    Off_Us b@ + b_etat b!
    then
    lon
    else
    drop
    drop
    drop
    drop ( 0 req b!
    then
    ;

i: OINT ( ... --- ...
( 0 INTO1cr b!
begin
    FFFF 01FFF0. @I - dup dup dup dup
    100 and 0< if 0 0 req #orb! Resc then
    200 and 0< if 1 1 req #orb! Resc then
    400 and 0< if 2 2 req #orb! Resc then
    800 and 0< if 3 3 req #orb! Resc then
    F00 and
    0 =
    until
    req b@ pontact b@ and 0 <
    if
    TB21cr b@ 8 or TB21cr b!
    then
    ( 3 INTO1cr b!
    ;i
    1 variable 0xxx 0 0xxx b!

```

(buffer réception plein ?

(établi la nature de la

(si tension

(établi le signe du courant

(enregistrement dans la

(lecture de la donnée

(lis le courant

(établi le signe de la tension

(enregistrement dans la

(lecture de la donnée

(lis la tension de sortie

(identifie le pont requérant

(si il y a une requête et le pont

(est actif alors on lance le programme principal

(il met a un la demande d'interruption

```

i: 1INT ( ... -- ...

                                ( ** protection de surcharge de courant
                                ( apparaît a la suite d'un courant qui dépasse
                                ( In courant nominal

0xxx b@ 10 >
if
(      stop_procedure
then
;i

: delais_cour ( n --- ...

                                ( reçoit la valeur du courant
                                ( calcule le temps de fonctionnement admissible
                                ( pour ce courant anormal
                                ( step = no. de cycles de décompte du timer courant
                                ( 1 cycle ~ 2 sec.
                                ( il choisit pour step la valeur minimale
                                ( entre celle ancienne et celle calculée

dup
blmnom b@
>

                                ( 10 -[Isortie-150A]/5 ->step 64hex--> 100A
                                (
                                ( fenêtre courant anormal avec indication du courant

                                if
                                0,00001 30 send
                                0 29 tran bl
                                1 30 tran bl
                                96 - 5 / negate A +
                                0 max step b@
                                min step bl

                                ( choisit le plus petit celui qui est reste
                                ( ou celui qui a été calculé
                                ( début timer courant
                                ( ** step doit être mis a la valeur max
                                ( seulement a la fin du
                                ( décompte / avec arrêt du système /
                                ( ou quand le courant revient au normal

                                1 CntSt #orbl

                                4 clIn

                                ( si le courant est revenu au normal
                                ( on met step a la valeur maximale- en attente

                                else
                                drop FF step bl
                                0, 29 send
                                0 30 tran bl
                                1 29 tran bl
                                4 cled #clrb!
                                4 P5 #clrb!
                                4 mask #clrb!

                                ( éteint le led rouge courant anormal

                                ( annule le bit qui empêche de clignoter le led
                                ( ." courant normal maître " cr

                                then

                                : ddelais_cour ( n --- ...

                                ( reçoit la valeur du courant
                                ( calcule le temps de fonctionnement admissible
                                ( pour ce courant anormal
                                ( step = no. de cycles de décompte du timer courant
                                ( 1 cycle ~ 2 sec.
                                ( il choisit pour step la valeur minimale
                                ( entre celle ancienne et celle calculée

                                dup
                                blsnom b@
                                >

                                ( 10 -[Isortie-150A]/5 ->step 64hex--> 100A

```

```

if
    0, 2A send
    0 2C tran b!
    1 2A tran b!
        96 - 5 / negate A +
        0 max sstep b@
        min sstep b!

    6 CntSt #orb!

    5 clin

else
    drop FF sstep b!
    0, 2C send
    1 2C tran b!
    0 2A tran b!
    5 P5 #clrb!
    5 cled #clrb!
    5 mask #clrb!

then
;
1 variable go
0 go b!
i: 2Int ( ... -- ...
    code
    lm lda,
    lma sta,
    Ad0 lda,
    lm sta,
    Ad3 lda,
    Si sta,
    Ur lda,
    Ura sta,
    Ad1 lda,
    Ur sta,
    Forth
go b@ 0 >
if
    FF P5 b!
    ." stop all " cr cr
    stop_procedure
then
( 0 INT1lcr b!
    TB2lcr b@ 8
    or TB2lcr b!
( F pontact b@
( and req b! ( lance le programme principal seulement pour les ponts actifs
;
i: timer_courant ( ... ____ ...

    step b@ dup
    0 =
    if
        (
        ( fenêtre courant anormal avec indication du courant

        ( choisit le plus petit celui qui est reste
        ( ou celui qui a été calculé
        ( début timer courant
        ( ** step doit être mis a la valeur max
        ( seulement a la fin du
        ( décompte / avec arrêt du système /
        ( ou quand le courant revient au normal

        ( si le courant est revenu au normal
        ( on met step a la valeur maximale- en attente

        ( éteint le led rouge courant anormal
        ( annule le bit qui empêche de clignoter le led
        ( ." courant normal esclave " cr
        ( fenêtre ok

        ( il lit le courant de sortie et la tension du réseau
        ( sauve l'ancienne valeur du courant

        ( lit la nouvelle valeur

        ( lit le signe du courant

        ( sauve l'ancienne tension du réseau

        ( lit la tension du réseau

        ( go <=> 0 active la protection du système

        ( on l'interdit jusqu'à ce que on va le servir

        ( il lance le programme principal

        ( ** protection de courant qui varie de façon linéaire
        ( avec le courant
        ( afin de respecter le S.O.A. du thyristor
        ( step = combien de fois est répète l'intervalle de 2 sec.

```

```

1 CntSt #clrb!
drop
stop_procedure
0, 25 send
1 25 tran b!
4 mask #orb!
4 clin
( ." surcharge longue durée maître " cr
( fenêtre surcharge de longue durée

else
  lm b@ lma b!
  Ad0 b@ dup lm b!
  blmnom b@
  >
  if
    1 - step b!
  else
    1 CntSt #clrb!
    FF step b!
    0, 29 send
    0 30 tran b!
    1 29 tran b!
    4 cled #clrb!
    4 P5 #clrb!
    4 mask #clrb!
    then
      then
        ;i
        i: timer_courant_esc
        ( ... --- ...
        ( ** protection de courant qui varie lineairement
        ( avec le courant
        ( afin de respecter le S.O.A. du thyristor
        ( step = combien de fois est répété l'intervalle de 2 sec.

        sstep b@ dup
        0 =
        if
          6 CntSt #clrb!
          drop
          stop_procedure
          0, 2B send
          0 2C tran b!
          1 2B tran b!
          5 mask #orb!
          5 clin
          ( ." surcharge longue durée esclave " cr
          ( fenêtre surcharge de longue dur/e

          Off_Is b@
          pontc b@ +
          b_etat b@
          blsnom b@
          >
          if
            1 - sstep b!
          else
            6 CntSt #clrb!
            FF sstep b!
            0, 2C send
            0 2A tran b!
            1 2C tran b!
            5 cled #clrb!
            5 P5 #clrb!
            ( empêche de clignoter la led
            ( la led va être toujours allumée
            ( diode: arrêt du système a cause de courant else

            ( dup . cr cr ." test " cr
            ( lit le courant de sortie après le filtre

            ( sstep b@ . ." sstep " cr

            ( remise a max du compteur de 2 sec.
            ( ." courant normal " ( fenêtre ok

            ( éteint le led rouge courant anormal

```

```

5 mask #clrb!                ( annule le bit qui empêche de clignoter le led
then
then
;
: initialize ( ... --- ...
    0 P7d b!
    E1 P6d b!
    FF P5d b!
    ( 3F P5d b!
    ( setAD
;
: norm ( ... --- ...
    ( oK avril 95
    ( ** protection Umax et Urmin
    ( ** établi le signe du courant et de la tension de sortie
    Im b@ lma b!                ( réactualise les donnée du maître
    Ad0 b@                      ( dup . ." Im PP " cr
    Im b!                        (
    Ur b@ Ura b!
    Ad1 b@                      ( dup . ." ur pp " cr
    Ur b!                      ( mesure de la tension du réseau
    Ur b@ dup                  ( tension de réseau valeur entière
    Ura b@ - ks b@ /           ( vérifie si il y a eu une variation
    0 <>                        ( ks = facteur de sensibilité
    if                          ( si il y a une variation il calcule
        dup dup s>f kUr f@ f*
        1 20 dup send tran b!    ( si oui envoi
        1 variation b!
        FE >                    ( protection Umax Urmin pour la tension du réseau
        if
            drop
        ( stop_procedure          ( tension réseau trop grande
            26 send 1 26 tran b!
            7 mask #orb!          ( il l'empêche de clignoter
            7 clin ( il l'allume
                                ( ." tension de réseau grande " cr cr cr
                                ( fenêtre Umaximal
                                ( tension réseau trop petite
        else
            dup Urmin b@
            <
            if
        ( stop_procedure
            s>f kUr f@ f*
            27 send 1 27 tran b!    ( fenêtre tension réseau trop petite
            7 clin
            7 mask #clrb!
        ( ." tension de réseau petite " cr cr cr
        ( 7 P5 #orb!
        else
            drop
        ( ." tens norm " cr cr cr
            0, 28 send
            0 26 tran b!
            0 27 tran b!
            1 28 tran b!
            7 cled #clrb!
            7 P5 #clrb!
            7 mask #clrb!
                                ( Éteint la diode si elle est reste allume
                                ( Arrête la minuterie de signalisation

```

```

        then
        then
        else
        drop
        then
        lm b@ dup lma b@ - ks b@ /
une variation
0 <>
if
    s>f klm f@ f* ( fdup f. cr
    Si b@ 0 <>
    if
        fneg
        then
        1 1F dup send tran bl
        lm b@ delais_cour
(
        1 variation bl
        else
        drop
        then
        Off_Us bget s>f kUs f@ f*
(
        Off_sus b@ 70 and 1 =
(
        if
(
            fneg
            then
            Off_Us pfsend
            Off_Is dup bget
            dup dup blsnom b@ >
            if
            pont b@ pontc bl
            ddelais_cour
            else
            drop
            0,9999 2C send
            ( ." norm else " cr
            0 2A tran bl
            1 2C tran bl
            5 P5 #clrb!
            5 cled #clrb!
            5 mask #clrb!
            then
            s>f kls f@ f*
            Off_sis b@ 70 and 1 =
            if
            fneg
            then
            pfsend
:
: varia ( ad1, ad2 --- ...
        ( ok 25 juillet 95
        ( pour les entiers
        ( Off-nouveau, Off-ancien, varia
( si il y a une différence il met le drapeau variation
( dans la matrice trans
bget
swap bget
- ks b@ / 0<>

```

(vérifie le courant du maître si il a subi
(ks = facteur de sensibilité = 1, 2, 4, 8
(calcul si il y a une variation
(établi le signe du courant du maître
(calcul le courant lu par le maître
(calcul le délais max de surcharge si
(c'est une surcharge
(établie les signes des tens. de sortie
(éteint le led rouge courant anormal
(annule le bit qui empêche de clignoter le led
(ajoute le signe au courant de sortie

```

if
    1 variation b!
then
:
:   fvaria ( ad1, ad2 --- ...
                ( vérifie les variables réelles
                ( Off-nouveau, Off_ancien, varia
( si il y a une différence il met le drapeau variation
fget
fget f- 0, f<>
if
    1 variation b!
then
:
:   verif ( ... --- ...
    ( tens_consigne_neuve a FF la tension de sortie est zéro
    ( les tensions et les courants de chaque pont seront
    ( mesurées [ 0 FF ] et transmises par les esclaves dans le protocole d'échange
    ( sur INTO
    (
    ( ** la routine calcule la tension de sortie moyenne;
    ( ** la variation de la consigne
    ( ** les variations des tensions des deux ponts par rapport a la tension moyenne de sortie
    ( les variables de différence ont le zéro a 7F
    ( les autres ont le zéro 0 hex
    ( delta_u = PID /consigne-tens_reseau/
    ( delta_i = PID/ I_point_travail - I_sortie/
    ( les coordonnées du tableau seront:
    ( u_tableau = u_poit_travail + delta_u
    ( I_tableau = I_poit_tavail + delta_i
    ( puis il cherche dans le tableau -> commande R0 R2
    ( envoi des commandes R0 + dR0 R2 + dR2
    Off_Us Off_Usa varia      ( vérifie si il y a eu lieu une variation
                                ( de la tension de sortie
    Off_Is Off_Isa varia      ( vérifie si il y a eu lieu une variation
                                ( du courant de sortie
    Off_C Off_Ca fvaria      ( vérifie si il a variation de la consigne
                                ( si le courant est plus grand que celui normal

    1 Off_C b@
    pont b@ + tran b!      ( oblige de l'afficher a l'écran
:
:   alpha
    c_initial @ -          ( vérifie le 3 août 1995
                                ( correction due au fait que le passage par zéro
                                ( ne correspond pas a celui de la tension
                                ( == il précède la tension
    s>f 19AD, f/          ( transforme la commande en radians
:
:   identif ( ... --- ...
                                ( identification du pont a régler
                                ( accorde une priorité décroissante:
                                ( pont 0, pont1, pont2, pont3
                                ( pont 3 sera réglé le dernier
                                ( si le ponts sont en parallèle ou série le réglage se fait
                                ( en utilisant les valeurs moyennes " pont équivalent 4 "

0 gen b!
gen b@ 0 <>
if
    4 pont b!              ( fonctionnement comme pont équivalent

```

```

else
  req b@
  dup 1 and 0<>
  if
    0 pont b!          ( 0 req #clrb!
    ( pont 0
  else
    dup 2 and 0<>
    if
      1 pont b!          ( 1 req #clrb!
      ( pont 1
    else
      dup 4 and 0<>
      if
        2 pont b! ( 2 req #clrb!
        ( pont 2
      else
        8 and 0<>
        if
          3 pont b! ( 3 req #clrb!
          ( pont 3
        then
          then
        then
      then
    then
  :
  : update ( ... -- ...
                                ( mise a jour de la donnée ancienne
swap fget pfsend
:
: mise_a_jour
( Ur b@ Ura b!
( Im b@ Ima b!
Off_Us Off_Usa update
Off_Is Off_Isa update
Off_C Off_Ca update
:
1 variable di 4 di b!
5 1 1 mat point
1 variable XOff_pt 10 XOff_pt b!

: command ( *** tient compte du sens du courant
           ( *** calcule la consigne_index pour chercher dans le tableau
           ( *** fait l'approximation linéaire pour la commande
           ( *** envoi les consignes aux esclaves
           ( ." commande vers les esclaves " cr
           ( le no. du pont qui va être commande se trouve dans la variable pont
( pont =0, 1, 2, 3, commandes des esclaves 0-3 et pont = 4 commande suivant ( le modèle du pont
équivalent

Off_C bget          ( initialisation pour tests
dup . cr di b@ /    ( consigne du pont
Off_Us bget di b@ / ( consigne fictive
-                   ( tension de sortie du pont
pont b@ point b@    ( Ucons-Usortie
+                   ( tens.point travail + variation_point 0 max 7F min
                   ( dup . ." pt "

```



```

    pont b@ point b!                ( on enregistre le point de travail
(   tens_reseau f@ fdup f. Urmin f@ fdup f. f-
(   f>d swap drop 0 max 1F min
    pont b@ point b@ 1 max 7E min    ( index matrice a partir de Ureseau 1D
                                      ( tens réseau fictive
        comm @ ( dup . ." Rp "
            Rp !                    ( commande approximative
            Rp @ ( dup . ." rp " cr
            67A max 67A - Rp !      ( correction pour décalage du passage par zéro
                                      Rp b@

        pont b@ 1 =
        if
            dup 0 >cp
        then
            ( dup .
( ***** commande utilisant boucles de tension indépendantes
(   pont b@ >cp ( LSB
( ***** fin section
( ***** commande utilisant une boucle de tension qui réglé les
( quatre esclaves
            dup 0 >cp'
            dup 1 >cp'
            dup 2 >cp'
            3 >cp'
( ***** fin section
            Rp 1 + b@
            3 max ( dup .
            pont b@ 1 =
            if
                dup 0 >cp
            then
( ***** commande utilisant boucles de tension indépendantes
(   pont b@ >cp' ( MSB
( ***** fin section
( ***** commande utilisant une boucle de tension qui réglé les
( quatre esclaves
            dup 0 >cp'
            dup 1 >cp'
            dup 2 >cp'
            3 >cp'
( ***** fin section
;
i: main          ( ... -- ...
                  ( interruption du programme principal
begin
req b@ pontact b@ and F and 0<>
if
    identif      ( identifie le pont a traiter
(   norm         ( protection tension réseau, signe courant maître
(   verif        ( vérifie si il y a des variations
    command      ( pour couper toute les boucles de réglage
    mise_a_jour

then
    0 req b!
pont b@ req #clrb!
req b@ 0=        ( répète la routine tant que toutes les
until            ( demandes ne sont pas satisfaites

```

```

( 2 INT2lcr bl      ( permet l'interruption de variation
2 INT0lcr bl
;
: ppont ( f -- f
( pont Graetz:  $3 \cdot \sqrt{6} / 2 \cdot \pi$  1,2B671AC f*
;
: Initvar
FF step bl
0 pont bl
0 idx bl
0 gen bl
0 req bl
( Offset commun pour etat et b_etat
1 Off_Us bl      ( tension de sortie
6 Off_Is bl      ( courant sortie
B Off_C bl       ( consigne
10 Off_Isa bl     ( courant ancien
( Offset seulement pour etat
1A Off_pt bl     ( point travail
33 Off_Ul bl      ( tension de ligne
( offset seulement pour b_etat
15 Off_sis bl     ( signe courant des esclaves
1A Off_sus bl     ( signe tension de sortie
1F Off_Usa bl     ( tension sortie ancienne
24 Off_Ca bl
30 Off_t bl
( offset pour d_etat
0 Off_R bl       ( consigne au pont
5 Off_dR bl      ( correction des consignes au pont
0 r bl
begin
0 Off_dR b@ r b@ + _etat !
r b@ 1 + r bl
r b@ 4 >
until
( ..... PONT 0 .....
0 dup . ." pont " cr pont bl
20, Off_C pfsend
Off_C fget f. ." C "
0, Off_Us pfsend
Off_Us fget f. ." Us "
0 Off_dR p_send
Off_dR gett . ." dR "
A, Off_Is pfsend
Off_Is fget f. ." Is "
5, Off_Isa pfsend
Off_Isa fget f. ." Isa " cr
1,2 Off_Z pfsend
Off_Z fget f. ." Z "
0 Off_R p_send
Off_R gett . ." R "
0, Off_pt pfsend
Off_pt fget f. ." pt " cr cr
( ..... PONT1 .....
1 pont bl
10, Off_C pfsend
1,2 Off_Z pfsend
10, Off_Us pfsend

```

```

0 Off_dR p_send
A, Off_Is pfsend
9, Off_Isa pfsend
0 Off_R p_send
10, Off_pt pfsend
Off_Us fget f. ." Us "
Off_dR gett . ." dR "
Off_Is fget f. ." Is "
Off_Isa fget f. ." Isa " cr
Off_C fget f. ." C "
Off_Z fget f. ." Z "
Off_R gett . ." R "
Off_pt fget f. ." pt " cr cr
2 dup . ." pont " cr pont bl
10, Off_C pfsend
Off_C fget f. ." C "
10, Off_Us pfsend
Off_Us fget f. ." Us "
0 Off_dR p_send
Off_dR gett . ." dR "
A, Off_Is pfsend
Off_Is fget f. ." Is "
F, Off_Isa pfsend
Off_Isa fget f. ." Isa " cr
1,2 Off_Z pfsend
Off_Z fget f. ." Z "
0 Off_R p_send
Off_R gett . ." R "
10, Off_pt pfsend
Off_pt fget f. ." pt " cr cr
( ..... PONT 3 .....
3 dup . ." pont " cr pont bl
10, Off_C pfsend
Off_C fget f. ." C "
10, Off_Isa pfsend
Off_Isa fget f. ." Isa " cr
10, Off_Us pfsend
Off_Us fget f. ." Us "
0 Off_dR p_send
Off_dR gett . ." dR "
A, Off_Is pfsend
Off_Is fget f. ." Is "
1,2 Off_Z pfsend
Off_Z fget f. ." Z "
0 Off_R p_send
Off_R gett . ." R "
10, Off_pt pfsend
Off_pt fget f. ." pt " cr cr
;
: setm
      initialize
(      initvar
      ' 0INT l0v l
      32 INT0lcr bl
01FFF0. @l pontact b@ 256* 16*
( F000
or 01FFF0. ll
      ' 1INT l1v l

```

(demandée par les esclaves
(sensible au niveau

(permet les interruptions des esclaves actifs
(variation courant de sortie

```

2 INT1 lcr bl
2 INT2 lcr bl          ( courant > courant maximal
' 2Int l2v l          ( start timer courant
D8 AdCtrl bl          ( ad repeat sweep mode
1 P5 bl              ( pour mesurer le courant il faut avoir le transistor off
                      ( Ad0 ----> courant sortie
0 Adlcr bl           ( Ad1 ----> tension du réseau
                      ( Ad3 ----> signe du courant
                      ( timer pour le courant
(
  C0 TA1m bl
  AF TA1 bl
  2 TA1 lcr bl
  ' timer_courant TA1v l
(
  C0 TA3m bl          ( pour TA3 timer clignotement courant
  FFF TA3 l          ( toutes les x sec.
  2 TA3 lcr bl
  ' tclin TA3v l
(
  44 TA4m bl          ( mise en route de l'horloge de 25 Khz pour les AOP.
  28 TA4 l
( 1 P6d #orbl
  4 CntSt #orbl
  ' main TB2v l
  1 TB2 lcr bl        ( programme principal
(
  C0 TB1m bl          ( pour TA3 timer surcharge esclaves
  FFF TB1 l          ( toutes les x sec.
  2 TB1 lcr bl
  ' timer_courant_esc TB1v l
(
  6 CntSt #orbl
  0 setlevel
;

: start              ( pour une tension de sortie nulle la consigne est 0
                    ( pour une tension de sortie maximale la consigne est 7F
(
  ( dec
  ( ." courant nominal de la charge : " finput courant_nominal f!
  ( ." courant maximal d'un thyristor : " finput courant_maximal f! ( ." tension nominale du réseau : " finput
  tens_sec f!
  ( calcul de la tension de sortie max fonction de la structure du redresseur
  ( et de la tension du réseau nominale => tens_sortie f!

  3A,      fdup f. ." tension de. réseau "
  cr      tens_reseau f!          ( 110v dec. tension de ligne du réseau
  44,      fdup f. ." Ur calibration " cr      Urcal f!
  44,      fdup f. ." Us calibration " cr      Uscal f!
  13,0 fdup f. ." l sortie max " cr      lsmx f!
  8,       fdup f. ." l sortie nominal " cr      lsn f!
  19,      fdup f. ." l esclave calibration " cr      lscal f!
  13,0 fdup f. ." l maître max " cr      lmmx f! 8, fdup f.
  ." l maître nominal " cr      lmn f!
  19, fdup f. ." l maître calibration " cr      lmcsl f!

( tension réseau
Urcal f@ FF, f/ fdup f. ." kUr " cr kUr f!          ( calcul du coefficient d'ecelle
                                                    ( pour lequel le système a été ajusté

tens_reseau f@ fdup 1,333333 ( Ureseau*1,2=Umax
f* fdup f. ." Urmx " fdup kUr f@ f/
f>d swap drop dup . ." bUrmx " cr bUrmx bl Urmx f!
0,CCCCC f* fdup f. ." Urmin " cr
fdup kUr f@ f/ f>d swap drop bUrmin bl Urmin f!

```



```

        until
        0 v_index !
        h_index @ 1+ ." ."
        dup h_index ! 1F >
        until
0 P5 bl
." fini " cr
then
( setm
;
: l command ;
: set_device ( ... -- ...
FF P5d bl      ( port 5 port de sortie E3 P6d bl
                0 P7d bl
;
1 variable idx 0 idx bl
1 variable vpar 0 vpar bl

1 vpar bl
( sousprogramme trpc
( envoi des données de la matrice de transmission qui ont changés au P.C.
l: trpc

        0 idx bl
        begin
            idx b@ etat f@ fdup 0,000001 f<
            if
                4 emit0 idx b@ . f. cr
            f_md idx b@ etat f!
            then
                1 idx b@ + idx bl
                idx b@ 1F >=
        until
;
: Intra ( ... -- ....
( initialisation de la transmission au P.C.
C0 TA0m bl
FFFF TA0 !
' trpc TA0v !
2 TA0lcr bl
0 setlevel
1 CntSt bl

0 idx bl
begin
3,453451 idx b@ etat f!
idx b@ etat f@ f.
1 idx b@ dup 1 + idx bl
17 >
until
0 idx bl
;

UpdTbl

```

(initialisation de la matrice
(appellation dans le programme: mm nn etat
(la donnée réelle mm calculé est stocké
(dans la position nn de la matrice
(drapeau de variation

(si oui reprend la sequence du debut
(debut boucle de transmission
(vérifie si il n'y a rien
(si il y a une donnée
(transmet : debut + index donnée +donnée
(0,000001
(remet ... zero

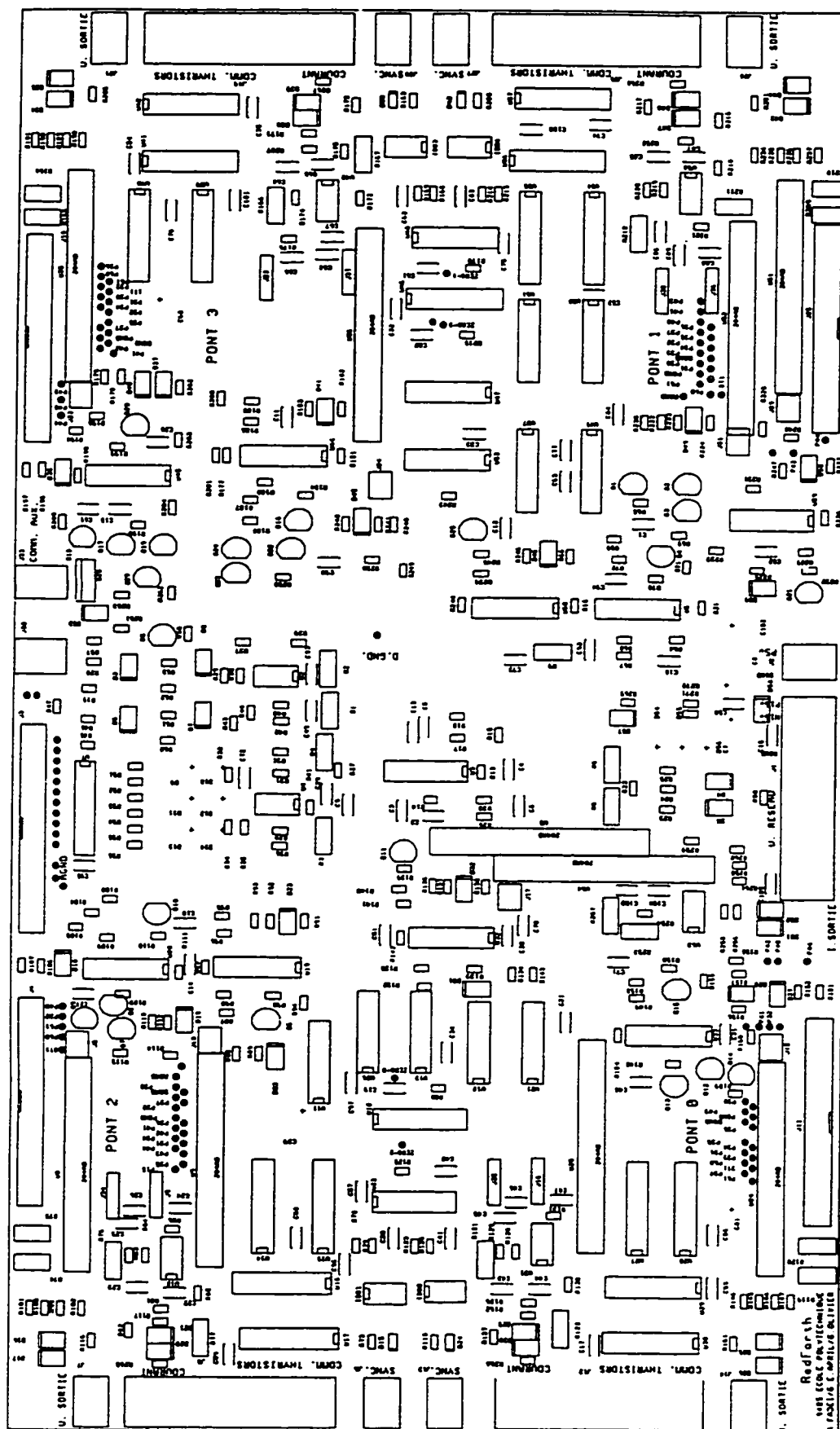
(incremente l'index
(vérifie si on a fini la matrice

(initialisation de la matrice de transmission

ANNEXE F

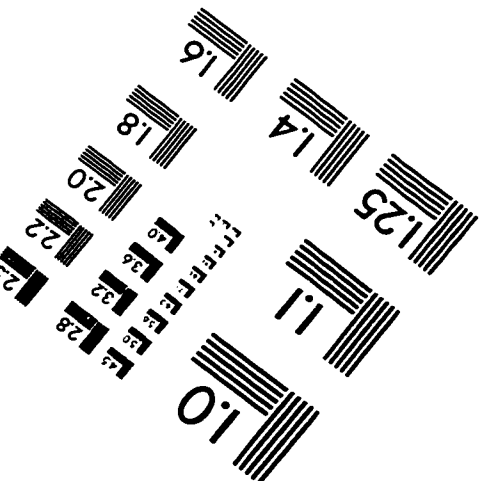
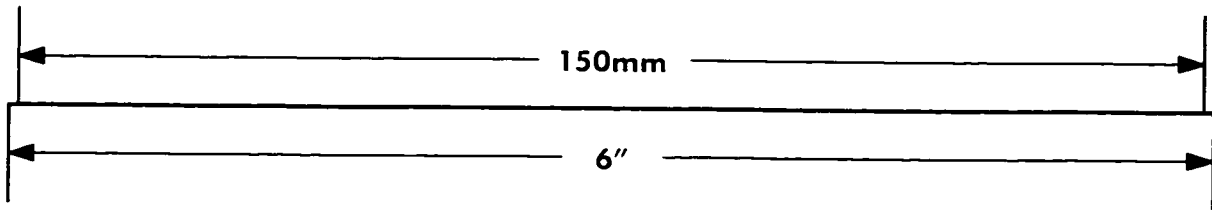
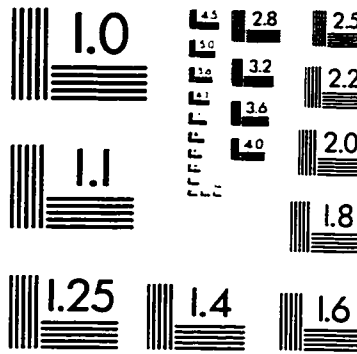
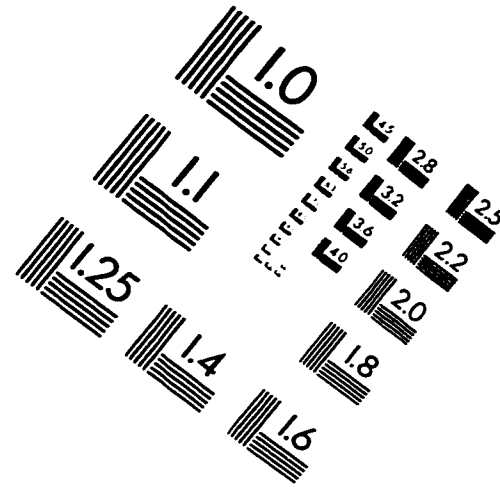
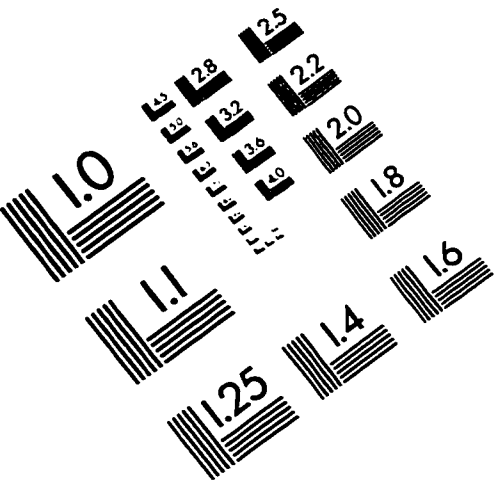
La carte d'interface entre le micro-ordinateur et l'étage de puissance a été conçue à l'aide des outils de développement de ORCAD. Les fichiers qui contiennent les schémas des circuits d'interface et le circuit imprimé de la carte Redforth sont archivés sur deux disquettes déposées avec ce mémoire sous le nom de M7. Le plan de la carte présente l'emplacement des composantes, l'identification des ponts commandés et la position des points de test pour les principaux signaux de la carte d'interface.

La carte est alimentée par des tensions de + 5 V et de +/- 15 V et les diodes luminescentes D54, D55, D56 indiquent leur présence. Il existe un seul point commun entre la masse de la carte d'interface et la masse du micro-ordinateur situé à la borne de masse de la source d'alimentation. Ainsi, on limite la propagation du bruit numérique vers les circuits analogiques de la carte d'interface.



PLAN DES COMPOSANTES DE LA CARTE REDFORTH.

IMAGE EVALUATION TEST TARGET (QA-3)



APPLIED IMAGE, Inc
1653 East Main Street
Rochester, NY 14609 USA
Phone: 716/482-0300
Fax: 716/288-5989

© 1993, Applied Image, Inc., All Rights Reserved

